НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису
УДК 004.942

До захисту допущено
В. о. завідувача кафедри ММСА
_____ О.Л.Тимощук
«___» _____ 2020 р.

Магістерська дисертація

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз
на тему: «Скорингові методи оцінки ризику для запобігання кредитного шахрайства»

Виконала:
студентка ІІ курсу, групи КА-91 мп
Зубко Марія Андріївна _____

Керівник: доцент кафедри ММСА
к.ф.-м.н, доц. Каніовська І.Ю. _____

Рецензент: доцент кафедри ММСА
к.ф.-м.н, доц. Ільєнко А.Б. _____

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів
без відповідних посилань
Студентка _____

Київ
2020

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти — другий (магістерський)
Спеціальність — 124 «Системний аналіз»

ЗАТВЕРДЖУЮ
В. о. завідувача кафедри ММСА

_____ О. Л. Тимощук
«___» _____ 2020 р.

ЗАВДАННЯ
на магістерську дисертацію студентці Зубко Марії Андріївні

1. Тема дисертації: «Скорингові методи оцінки ризику для запобігання кредитного шахрайства», науковий керівник дисертації Каніовська Ірина Юріївна, к.ф.-м.н., доцент, затверджені наказом по університету від «02» листопада № 3182-с

2. Термін подання студентом дисертації: 14 грудня 2020 р.
3. Об'єкт дослідження: Історичні дані по транзакціям юридичних осіб та інформацією про те, чи є клієнт шахраєм, чи ні.
4. Предмет дослідження: Методи прогнозування і побудови скорингової карти.
5. Перелік завдань, які потрібно розробити:
1) дослідити сучасний стан систем запобігання шахрайства та особливості застосування автоматичного скорингу;
2) розробити скорингову карту для оцінки ризикованості клієнтів за допомогою логістичної регресії;
3) створити програмний продукт для аналізу та запобігання кредитного шахрайства;
4) знайти дані для застосування в програмному продукті;
5) протестувати прогностичну силу розробленої моделі;
6) розробити стартап-проект виведення на ринок результатів дослідження;
7) зробити висновки за результатами наукового дослідження
6. Орієнтовний перелік графічного (ілюстративного) матеріалу:

1) Ілюстрації до математичних методів, що лежать в основі наукового дослідження (рис. 2.1 – рис. 2.5);
2) Результати роботи створеного програмного продукту (рис. 3.1 - рис. 3.37, табл. 3.1 – табл. 3.3);
3) Таблиці у розділі стартап-проекту (табл. 4.1 – табл. 4.22)

7. Орієнтовний перелік публікацій: участь у міжнародній науково-практичній конференції «Відкриті еволюціонуючі системи» з опублікуванням тез

8. Дата видачі завдання: 01 вересня 2020 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації
1.	Концептуальний вступ дисертації. Формулювання об’єкта, предмета, цілі, завдань, новизни, практичної значущості результатів	01.09.2020—10.09.2020
2.	Перший розділ. Огляд літературно-інформаційних джерел. Понятійно-категоріальний апарат. Характеристика об’єкта	11.09.2020—28.09.2020
3.	Другий розділ. Опис математичного апарату та методів, що застосовуються для досягнення поставленої цілі	28.09.2020—21.10.2020
4.	Третій розділ. Створення програмного продукту. Тестування програми	21.10.2020—15.11.2020
5.	Четвертий розділ. Стартап-проект	15.11.2020—20.11.2020
6.	Концептуальні висновки. Перспективи розвитку отриманих рішень	20.11.2020—25.11.2020

Студентка _____ М.А. Зубко
(підпис)

Науковий керівник дисертації _____ І.Ю. Каніовська
(підпис)

РЕФЕРАТ

Магістерська дисертація: 77 с., 4 ч., 25 табл., 42 рис., 1 дод., 16 джерел.

КРЕДИТНЕ ШАХРАЙСТВО, КРОС-ВАЛІДАЦІЯ, ЛОГІСТИЧНА РЕГРЕСІЯ, МЕТОД МОНОТОННОГО ОПТИМАЛЬНОГО РОЗБИТТЯ, СКОРИНГОВА КАРТА, ФРОД-СКОРИНГ

Об’єкт дослідження – історичні дані по транзакціям юридичних осіб та інформацією про те, чи є клієнт шахраєм, чи ні.

Предмет дослідження – методи прогнозування і побудови скорингової карти.

Мета дослідження – проаналізувати об’єкт дослідження, побудувати і протестувати моделі прогнозування, провести порівняння результатів.

Методи дослідження – метод розбиття змінних на групи: метод монотонного оптимального розбиття; метод прогнозування: логістична регресія та логістична регресія з використанням крос-валідації.

Актуальність – споживче кредитування, особливо експрес-кредитування передбачає практично миттєве рішення по видачі кредитів. В зв'язку з появою шахраїв, які мають на меті отримати гроші в банку і не повертати, а також постійним вдосконаленням їх дій, дуже важливо вміти швидко та якісно розпізнавати та попереджати ці дії.

Новизна – на відміну від ручного підходу, автоматизований скоринг дозволяє зекономити ресурси, унеможливити упередженість співробітників та помилки через людський фактор.

Результати дослідження – була побудована модель логістичної регресії, результати якої були покращені за допомогою крос-валідації та сформована скорингова карта.

ABSTRACT

The master thesis: 77 p., 4 s., 25 tabl., 42 fig., 1 appendix, 16 references.

CREDIT FRAUD, CROSS VALIDATION, LOGISTIC REGRESSION, METHOD OF MONOTONIC OPTIMAL BINNING, SCORECARD, FROD SCORING

The object of the study – historical data with transactions of legal entities and information on whether the client is a fraud or not.

The subject of the study – methods of forecasting and construction of a scorecard.

The purpose of the study – to analyze the object of study, build and test forecasting models, compare results.

Methods of the study - method of variables binning: method of monotonic optimal binning; forecasting method: logistic regression and logistic regression using cross-validation.

The relevance of the study – a consumer lending, especially express lending, provides an almost instant decision. Due to emergence of frauds who aim to get money in the bank and not to return it, as well as the constant improvement of their actions, it is very important to be able to quickly and efficiently recognize and prevent these actions.

Novelty - in contrast to the manual approach, automated scoring saves resources, eliminates employee bias and errors due to the human factor.

The results of the study – a logistic regression model was built, the results of which were improved by cross-validation and a scoring map was formed.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ 8

ВСТУП 9

РОЗДІЛ 1 АНАЛІЗ ІСНУЮЧОЇ ПРОБЛЕМИ 10

1.1 Процедури запобігання шахрайства як невід'ємний елемент управління кредитним портфелем 10

1.2 Переваги автоматизованого скорингу у проблемі запобігання кредитного шахрайства 12

1.3 Висновки 14

РОЗДІЛ 2 МАТЕМАТИЧНІ ОСНОВИ 15

2.1 Основні поняття 15

2.2 Алгоритми розбиття 19

2.3 Алгоритм монотонного оптимального розбиття 21

2.4 Логістична регресія 23

2.5 Крос-валідація 26

2.6 Тестування моделі 27

2.7 Вигляд скорингової карти 30

2.8 Висновки 32

РОЗДІЛ 3 АНАЛІЗ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ 33

3.1 Розрахунок характеристик 33

3.2 Застосування алгоритму розбиття 34

3.3 Побудова і тестування логістичної регресії 46

3.4 Пошук найкращої моделі за допомогою крос-валідації 49

3.5 Висновки 52

РОЗДІЛ 4 СТАРТАП ПРОЕКТ «АНТИФРОД» 54

4.1 Опис ідеї проекту 54

4.2.Технологічний аудит ідеї проекту 55
4.3 Аналіз ринкових можливостей запуску стартап-проекту 56
4.4 Розроблення ринкової стратегії проекту 63
4.5 Розроблення маркетингової програми стартап-проекту 66
4.6 Висновки 69
ВИСНОВКИ 70
ПЕРЕЛІК ПОСИЛАНЬ 71
ДОДАТОК А ЛІСТИНГ ПРОГРАМИ 73

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

WOE – вага доказу (weight of evidence)

IV – інформаційний індекс (information value)

ММП – метод максимальної правдоподібності

BFGS – алгоритм Бройде - Флетчера - Гольдфарба – Шанно (Broyden–Fletcher–Goldfarb–Shanno)

ROC – робоча характеристика приймача (receiver operating characteristic)

AUC – площа під кривою (area under curve)

PSI – індекс стабільності популяції (population stability index)

FICO – компанія Фера та Ісаака (Fair, Isaac and Company)

HI – індекс Херфіндаля (Herfindahl index)

ПДВ – податок на додану вартість

ВСТУП

В рамках даної роботи пропонується та досліджується автоматичний варіант запобігання кредитного шахрайства з використанням скорингових методів. Попередження шахрайства стає все більш необхідним в сучасних реаліях. Для кожної фінансової установи важливо не тільки оцінювати кредитоспроможність позичальників, а й вміти розпізнавати і попереджати шахрайські дії.

В більшості випадків дана проблема вирішується у ручному режимі, що потребує багато часу та ресурсів, а також припускає можливі помилки пов’язані з людським фактором. Запропонований підхід дозволяє значно підвищити якість кредитного портфеля, зменшуючи навантаження на верифікаторів та скорочуючи об’єм необхідних ресурсів.

Робота складається з чотирьох розділів.

У першому розділі описані особливості предметної області, існуючі підходи та переваги обраного методу.

Другий розділ присвячено опису математичних методів для вирішення проблеми запобігання шахрайства. В ньому описані основні підходи до розбиття вхідних характеристик, тести та показники, що використовуються для оцінки міри їх прогностичної сили, методи прогнозування та тестування моделей, формальний вигляд скорингової карти.

Третій розділ містить покроковий опис обробки вхідних даних, створення та дослідження характеристик, застосування на них алгоритму розбиття, виявлення залежностей цих характеристик із прогнозованою змінною, побудови скорингових моделей, їх тестування та порівняння.

У четвертому розділі наведений опис стартап-проекту.

РОЗДІЛ 1 АНАЛІЗ ІСНУЮЧОЇ ПРОБЛЕМИ

1.1 Процедури запобігання шахрайства як невід'ємний елемент управління кредитним портфелем

Кредитний портфель - це сукупність залишків заборгованості за основним боргом по активних кредитних операціях на певну дату.

Клієнтський кредитний портфель є його складовою частиною і являє собою залишок заборгованості за кредитними операціями банку з фізичними та юридичними особами на певну дату.

Серед традиційних видів банківської діяльності надання кредитів є основною операцією, що забезпечує їх прибутковість і стабільність існування. Видаючи кредити фізичним та юридичним особам, банк формує свій кредитний портфель. На фактичний стан клієнтського кредитного портфеля впливає прийнята банком система управління.

Управління кредитним портфелем являє собою організацію діяльності банку при здійсненні процесу кредитування, яка спрямована на запобігання або мінімізацію кредитного ризику. Кінцевими цілями кредитної організації при управлінні кредитним портфелем є, по-перше, отримання прибутку від активних операцій, по-друге - підтримку надійної та безпечної діяльності банку.

Жорсткість вимог з боку Центрального Банку і нестабільні економічні умови, в яких доводиться функціонувати фінансовим інститутам, зумовили підвищений інтерес кредитних установ до збереження якості портфеля, мінімізації кредитних ризиків і зниження обсягів проблемних активів.

Однією з фундаментальних завдань для досягнення цих цілей є налагодження ефективної роботи процедур запобігання шахрайства, яка може сприяти помітному зміцненню фінансової та репутаційної стабільності кредитного закладу. Такий напрям діяльності можна віднести до елементів

кредитної політики, характерним для поточного етапу економічного розвитку.

До поточних проблем комерційних банків (проблем позичальників) відноситься скорочення кількості позичальників з гарною кредитною історією та фінансовим станом, а також загальна недовіра громадян до банків. Метою позичальника в даному випадку є отримання необґрунтованих переваг при отриманні кредиту, іноді явне неповернення.

Відносини банку з позичальником починаються з письмового звернення - кредитної заявки, і перш ніж буде укладено кредитний договір, клієнт проходить процедуру кредитного аналізу. Для таких цілей апарат світової фінансової аналітики активно використовує андеррайтинг.

Андеррайтинг має кілька значень в фінансовому секторі, одне з них - оцінка ризиків при прийнятті рішення про надання кредиту або при укладанні будь-якого іншого договору.

Кожен банк має свою власну систему аналізу позичальника. За результатами перевірки, банк або дає свою згоду на видачу кредиту, або відмовляє в цьому. Кредитна організація може також прийняти рішення про надання позики не на тих умовах, які запитував клієнт. Наприклад, банк може зменшити суму кредиту і / або збільшити процентну ставку.

Можна виділити два типи андеррайтингу: автоматичний (скоринг) та індивідуальний. Для оцінки кредитоспроможності клієнтів використовується автоматичний скоринг, а от для виявлення підозрілих дій та перевірки підозрілих даних в більшості випадків застосовується індивідуальний андеррайтинг та глибокий аналіз діяльності клієнта. Остаточне рішення по заявці в цьому випадку виносить андеррайтер, який аналізує інформацію, надану клієнтом і суміжними службами. Даний тип обробки може займати від 1 до 10 днів.

У разі роздрібного беззаставного кредитування типовість позичальників і пропонованих кредитних договорів досить висока, тому набувають значимість такі механізми оцінювання потенційних і поточних

позичальників, які здатні обробляти великі обсяги інформації та класифікувати оцінювані об'єкти по заданим принципам. Важливо впроваджувати нові аналітичні, а головне автоматизовані інструменти, що дозволяють виявити кредитних шахраїв ще на рівні заявки. Статистичні методи аналізу даних можуть дозволити підвищити ефективність роботи з виявлення нестандартних дій.

Актуальність пов'язана зі споживчим кредитуванням, особливо з експрес-кредитуванням, яке передбачає практично миттєве рішення по видачі кредитів. Це стає особливо важливим у зв'язку з появою не тільки шахраїв-одинаків, але й шахрайських груп, які мають на меті отримати гроші в банку і не повертати. Шахрайські дії постійно вдосконалюються, тому дуже важливо вміти розпізнавати та попереджати ці дії [1].

1.2 Переваги автоматизованого скорингу у проблемі запобігання кредитного шахрайства

Автоматизованим варіантом вирішення проблеми запобігання кредитного шахрайства є модель скорингу, яка дозволяє визначити ризик шахрайства на першому етапі обробки кредитної заявки.

Вважається, що до 10% неповернень по кредитах пов'язані саме з відвертим шахрайством і цей показник зростає. Тому система запобігання кредитного шахрайства має бути в кожному банку в тому і чи іншому вигляді. Причому власну програму і критерії оцінки потенційних шахрайських дій кожна кредитна організація вважає своєю комерційною таємницею.

Класична модель прийняття рішень щодо видачі позики, як правило, включає в себе тільки скоринг на етапі заявки, завдання якого полягає в тому, щоб оцінити ймовірність дефолту по клієнту та прийняти рішення по заявці

на позику. Фрод-скоринг допомагає визначити ймовірність того, що клієнт є шахраєм.

Фрод-скоринг – це такий вид скорингу, що за допомогою статистичних методів допомагає оцінити ймовірність шахрайських дій з боку потенційного позичальника. Він використовується перш за все як певний бар'єр на шляху шахраїв при отриманні кредиту. Його функція – це перевірка наданих даних на протиріччя [2].

Даний підхід може бути використаний у випадку, коли банк пропонує чи погоджує кредити для клієнтів, що мають транзакційну історію в цьому банку, що дозволяє дослідити поведінку цих клієнтів у минулому та зробити висновки щодо адекватності їх поведінки чи особливостей ведення бізнесу у випадку з юридичними особами.

Автоматичний скоринг допомагає знизити навантаження на верифікацію. Після відсікання на початковому етапі заявка не проходить подальший цикл кредитного конвеєра, отже, зникає необхідність запитувати додаткові дані, що теж скорочує витрати.

При побудові фрод-скорингу необхідно враховувати, що шахраї – це люди, які розуміють принципи роботи звичайного скорингу. Тому заявки від шахраїв часто виглядають занадто ідеальними. За звичайним скорингом такі клієнти набирають хороший бал, і у більшості компаній подібні заявки перейдуть до автоматичного схвалення, пропустивши верифікацію. Тому фрод-скоринг повинен оцінювати такі заявки по-іншому й звертати увагу на інші нюанси, які не враховувалися під час звичайного скорингу.

Таким чином, спочатку ми вирішуємо, кому взагалі можемо видати позику й вже після цього звичайний скоринг допомагає розібратися, чи потягне позичальник боргове навантаження.

Схема кредитного конвеєра полягає в тому, що верифікатори обробляють тільки «сірий» сегмент, коли машина не справляється з прийняттям рішення. Впровадження фрод-скорингу дозволяє звузити «сіру» зону в кілька разів. Очевидно, що з введенням фрод-скорингу верифікатори

мають обробляти лише малу частину потоку в тих випадках, коли заявка виявилася підозрілою за результатами скорингу.

Впровадження фрод-скорингу має значний потенціал, тому що зараз з'являється все більше різних джерел даних, і це найкращий момент, щоб вийти за межі існуючих правил.

Фрод-скоринг допомагає також вивчати краще кредитний портфель. Розпізнаючи сумлінних клієнтів, ми можемо управляти їх кредитним навантаженням через аналіз їх фінансового стану на момент звернення в компанію, наприклад, за допомогою альтернативної пропозиції.

Отже, обраний автоматичний підхід має такі переваги:

а) економія часу і фінансових витрат;

б) швидке прийняття рішення;

в) відсутність упередженості співробітників по відношенню до позичальника;

г) унеможливлення помилки через людський фактор;

д) виявлення цікавих залежностей у поведінці клієнта, що можуть бути використані в подальшому.

1.3 Висновки

У першому розділі було описано важливість такої складової аналізу та керування якістю кредитного портфеля як процедури запобігання шахрайства. Дана проблема стає все більш актуальною в сучасних умовах і потребує окремої уваги. Більшість фінансових установ використовує ручну перевірку адекватності поведінки позичальників, що потребує великих ресурсів і може допускати помилки через вплив людського фактора, саме тому було знайдено та описано суть підходу для вирішення цієї проблеми – фрод-скорингу.

РОЗДІЛ 2 МАТЕМАТИЧНІ ОСНОВИ

2.1 Основні поняття

У машинному навчанні та аналітиці значна кількість завдань зводиться до бінарної класифікації. Кількість прикладних проблем, які можуть бути сформульовані в термінах «подія» та зворотному йому «не-подія», досить велике. Тому побудова ефективних бінарних класифікаторів є актуальною та важливою практичною задачею. Одним з поширених шляхів підвищення їх якості є розробка нових алгоритмів і модифікації існуючих. Згодом часто виявляється, що найбільш ефективні алгоритми (нейронні мережі, ліс рішень) погано інтерпретуються дослідником, що узгоджується з принципом невизначеності: чим вище точність, тим гірше інтерпретація (рис. 2.1). А інтерпретація є досить важливим фактором для банків та інших фінансових установ, адже процес погодження кредитів має бути прозорим та кожне рішення повинно мати чітке обґрунтування. Тобто важливо знайти рівновагу між точністю та складністю.

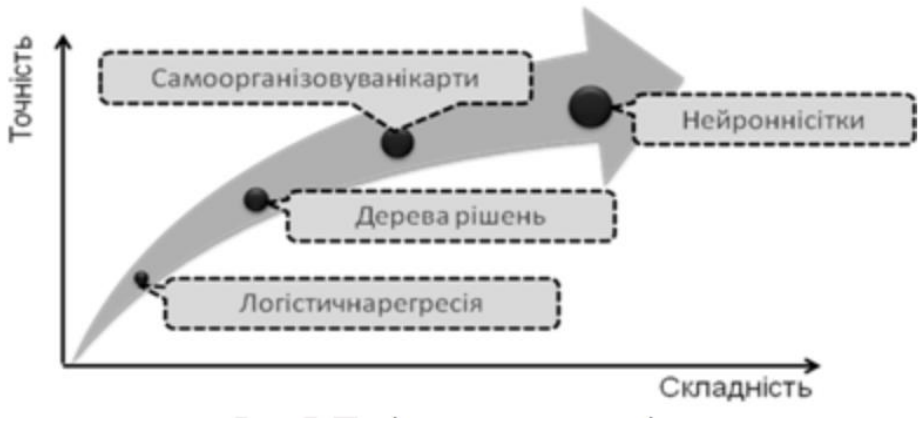

Рисунок 2.1 – Принцип невизначеності

Існує ряд підходів, що дозволяють провести попередню обробку навчальних вибірок з метою поліпшення роботи класифікаторів, а також вирішити ряд супутніх завдань: дослідити значимість вхідних змінних, і, в тій або іншій формі, перевірити гіпотези про причинні зв'язки між ними.

Одним з таких підходів є розбиття змінних на групи. В англомовній літературі застосовується термін «Fine & Coarse Classing», що можна перекласти як «початкові і кінцеві класи» [3].

Процедура формування кінцевих класів визначає, як буде представлена змінна в вибірці з точки зору кількості її унікальних значень. По суті, це є скорочення числа різноманітних значень змінної, яке зазвичай пов'язують зі зміною інтервалу дискретизації значень. Завдання полягає в зменшенні кількості значень вихідного набору даних за рахунок їх об'єднання в межах деякого інтервалу з використанням інформації про цільову змінну. В результаті такого перетворення кількість значень змінної має зменшитися без істотного збитку для інформативності даних.

Розглянемо докладно постановку задачі на прикладі інтервального змінної *x*, що приймає значення на діапазоні [L, R]. Опишемо її як n-мірний вектор початкових значень, або початкових класів, які зустрічалися в навчальній вибірці:

$$\bar{a} = \{a_j \in Z: a_j < a_{j+1}, a_0 = L, a_{n-1} = R, j = [0; n-1]\}.$$

Даний діапазон розбивається на m інтервалів квантування, $m < n$, границями яких можуть бути тільки початкові класи:

$$[L, R] = \bigcup_{k=0}^{m-2} (b_j, b_{j+1}],$$

$$b_k \in \{a_j\}, b_0 = L, b_{m-1} = R.$$

Кінцеві класи однозначно визначаються m-мірним вектором границь інтервалів квантування:

$$\bar{b} = \{b_k \in \{a_j\}: b_k < b_{k+1}, b_0 = L, b_{m-1} = R, k = [0; m-1]\}.$$

Спосіб пошуку вектору $\bar{b}$ визначає той або інший алгоритм формування кінцевих класів [4, 5].

Щоб відокремити хороші рахунки від поганих введемо поняття коефіцієнту WoE, або ваги доказу.

Нехай є навчальна вибірка, в якій кожному об'єкту, який описується набором змінних, ставиться у відповідність бінарна цільова змінна класу з двома станами - подія і не-подія. Для довільного інтервалу $[b_k; b_{k+1}]$ обчислюється коефіцієнт WoE:

$$WoE = ln\left(\frac{N_k/N}{P_k/P}\right) = \ln\frac{F^-}{F^+},$$

де k - індекс початкового класу, N_k - кількість не-подій, що потрапили в інтервал, N - загальна кількість не-подій у вихідному наборі даних, P_k - кількість подій, що потрапили в клас, P - загальна кількість подій.

Інтерпретація коефіцієнтів WoE наступна. У чисельнику під логарифмом відносна частота появи не-подій в класі F^-, а в знаменнику - відносна частота появи подій F^+. Якщо $F^- > F^+$, то логарифм їх відношення також більше 0 (рис. 2.2).

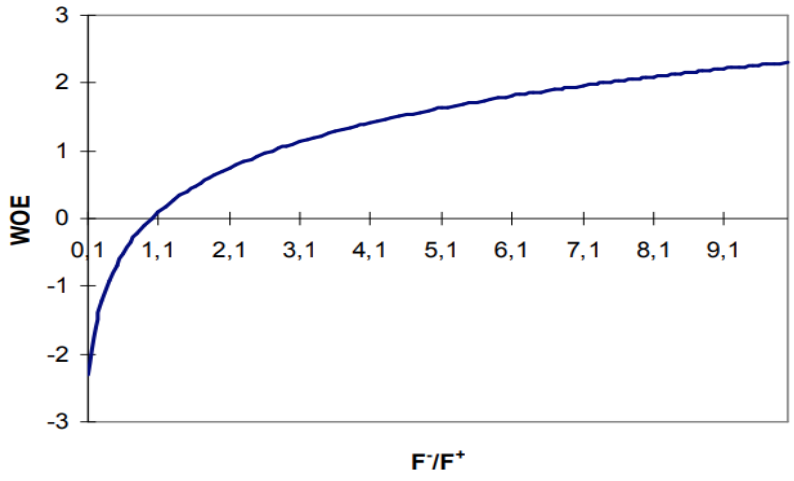

Рисунок 2.2 – Логарифмічна залежність для показника WoE

Негативні значення коефіцієнтів WoE вказують на велику ймовірність появи подій в інтервалі [6].

На основі коефіцієнтів WoE обчислюється величина, яка визначає значимість змінної, що називається інформаційним індексом, за формулою:

$$IV = \sum_{k=0}^{m-1} \left\{ \left(\frac{N_k}{N} - \frac{P_k}{P} \right) * WoE_k \right\}.$$

Інформаційний індекс пропорційний різниці відносних частот появи подій і не-подій в кожному інтервалі, підсумувавши по всіх інтервалах. Інформаційний індекс завжди є позитивною величиною і чим він більший, там більш значимою є змінна [7, 8].

Отже, суть підходу полягає в тому, аби тренувати модель не на початкових значеннях незалежної змінної, а на присвоєних відповідно до попадання цих значень у визначений інтервал значень WoE.

Перевагами такого підходу є:

а) підвищення стабільності моделі: деякі значення змінних можуть траплятися рідко, і якщо не згрупувати їх, це призведе до нестабільності;

б) покращення якості: групування подібних значень із подібними силами прогнозування збільшить точність прогнозування;

в) дозволяє зрозуміти логічні тенденції «хороших / поганих» клієнтів для кожної характеристики;

г) запобігає погіршенню змінних, яке можливе через екстремальні значення;

д) запобігає можливому перетренуванню.

2.2 Алгоритми розбиття

Алгоритми розбиття мають слідувати таким правилам:

а) пропущені значення змінної мають відділятися в окрему групу;
б) кожен інтервал повинен містити щонайменше 5% відсотків від усієї кількості спостережень змінної;
в) жодні інтервали не повинні мати 0 спостережень з не-подією, або 0 спостережень з подією [9].

Існують такі алгоритми розбиття:

а) Рівна ширина - це очевидний і прямий підхід до розбиття. Цілий діапазон значень незалежної змінної ділиться на заздалегідь задану кількість інтервалів однакової ширини.
б) Рівний розмір – підхід до розбиття, коли цілий діапазон значень незалежної змінної ділиться таким чином, що сформовані в результаті інтервали містять більш-менш однакову кількість спостережень. Ширина інтервалів варіюється залежно від щільності спостережень. Цільова кількість інтервалів заздалегідь визначена.

   Ці два алгоритми не використовують взаємозв'язок з цільовою змінною.
в) Оптимальне розбиття - це еволюція попередніх двох алгоритмів. В цьому випадку використання залежної змінної використовується для визначення граничних точок для інтервалів. Метою алгоритму є визначити, що інтервали мають достатньо різні статистичні середні оцінки значення незалежної змінної. Алгоритм складається з наступних кроків:
   1) починаємо з досить великої кількості маленьких інтервалів;
   2) для кожної сусідньої пари інтервалів обчислюємо *p*-значення, використовуючи Z-тест, що показує наскільки

статистично відрізняються частки подій для кожного інтервалу;

3) знаходимо найбільше значення *p* з усіх пар. Якщо це значення вище заданого порогу, то відповідна пара інтервалів зливається та алгоритм повторюється з кроку b), інакше закінчується.

г) Максимальна ймовірність однотонності - грубий класифікатор також відомий під назвою "Монотонний суміжний алгоритм об'єднання". Припустимо, що частка «не-подій» знижується зі збільшенням значення незалежної змінної. Починаємо з самого низького значення та продовжуємо додавати значення до тих пір, поки сукупна частка «не-подій» не досягне свого максимуму. Це перша точка розбиття. Потім обчислюємо сукупну частку «не-подій» з цього моменту, поки вона знову не досягне максимуму. Це є друга точка розбиття. Повторюємо процес поки не покриємо усі значення змінної.

д) Багатоінтервальна дискретизація базується на евристичному пошуку та мінімізації ентропії шляхом рекурсивного розбиття безперервного діапазону на підінтервали та рекурсивного визначення найкращих інтервалів.

е) Хі-злиття – це алгоритм дещо схожий на оптимальний. Різниця в тому, що критерій Хі-квадрат використовується для перевірки схожості суміжних інтервалів.

ж) Умовні дерева – це алгоритм заснований на вичерпному пошуку, який дозволяє збільшити обрану статистику.

2.3 Алгоритм монотонного оптимального розбиття

Що стосується оцінки ризику, ми маємо деякі специфічні вимоги до алгоритму розбиття. Можна виділити такі три вимоги:

а) Монотонність: очікується, що алгоритм розділяє вхідний набір даних на інтервали таким чином, що якщо йти від одного інтервалу до іншого в тому ж напрямку відбувається монотонна зміна ризику. Це випливає з основного припущення, що незалежна змінна здатна відокремити вищі ризики від менших ризиків у випадку глобальних немонотонних відносин.

б) Репрезентативність: очікується від алгоритму для налаштування інтервалів таким чином, щоб вони відображали максимальну кореляцію між незалежними змінними та показником ризику. Потім результати для кожного інтервалу можуть бути використані для побудови та тренування моделі регресії.

в) Обмеження: від алгоритму очікується виконання всіх умов описаних у попередньому підрозділі.

Алгоритм максимальної ймовірності однотонності побудований навколо першої вимоги. Однак це створює інтервали, які не є статистично важливими, із занадто малою кількістю спостережень, що скоріше розглядаються як викиди або статистичний шум. Це завжди відбуватиметься з реальними даними, які містять рідкісні ділянки та ненормальні викиди.

Алгоритми, такі як Хі-злиття та оптимальне розбиття, розроблені навколо другої вимоги - максимальної кореляції між незалежною змінною та залежним показником, вони спрямовані на побудову інтервалів, розділених за дисперсією чи точковою оцінкою (*p*-значення).

Алгоритми багатоінтервальної дискретизації та умовних дерев потрапляють в одну категорію, вони використовують ентропію інформації та умови для побудови більш когерентних інтервалів.

Прості алгоритми, такі як однаковий розмір або однакова ширина, мають перевагу у створенні обмеженої кількості інтервалів, що є третьою вимогою для досконалих алгоритмів розбиття [10].

Запропонований метод враховує всі три вимоги. Він поєднує переваги оптимального розбиття за значенням p та монотонні алгоритми грубого класифікатора в єдиному алгоритмі [11, 12]. Він також використовує монотонні відносини між незалежною змінною та індикатором і складається з наступних кроків:

а) Відсортувати дані за значенням залежної змінної, розділити на найменші можливі інтервали.

б) Монотонність: починаючи з верхнього інтервалу, пройти до останнього і об'єднати їх, якщо частка подій більша, ніж у наступного інтервалу, а потім перезапустити прохід зверху. Якщо злиття не відбудеться - перейти до наступного кроку.

в) Оптимальність за значенням p: для кожної сусідньої пари інтервалів обчислити значення p за допомогою критерію Хі-квадрат аби подивитися наскільки статистично відрізняються частки подій для кожного інтервалу.

У статистичному критерії гіпотеза H_0: випадкова величина підкорюється закону розподілу F(x), статистика:

$$\chi^2 = \sum_{j=1}^{k} \frac{\left(O_j - E_j\right)^2}{E_j} \sim {\chi^2}_{k-1},$$

де O_j – спостережувана частота подій, E_j – очікувана частота подій.

г) Фільтрування за розміром інтервалу: для кожної пари інтервалів, якщо будь-який із двох має розмір менше, ніж заздалегідь визначений поріг, змінити відповідне значення p, збільшивши його на одиницю.

д) Фільтрування за найменшою часткою подій: для кожної пари інтервалів, якщо будь-який із двох має частку, нижчу за деякий заздалегідь визначений поріг, змінити відповідне p значення, збільшуючи його на одиницю.

е) Знайти найбільше модифіковане значення p, якщо всі p-значення менше зазначеного порогу - вийти з алгоритму.

ж) В іншому випадку об’єднати два інтервали, що відповідають найбільшому модифікованому значенню p.

з) Перейти до кроку 3.

2.4 Логістична регресія

Не дивлячись на різноманітність алгоритмів побудови бінарних класифікаторів, логістична регресія на сьогодні залишається популярним інструментом, так як дозволяє отримувати добре інтерпретовані скорингові карти та ймовірності оцінки подій.

Логістична регресія або логіт-модель - це статистична модель, що використовує для прогнозування ймовірностей появи деякої події порівняння з логістичною кривою. Ця регресія видає відповідь у вірогідності бінарної події (1 або 0).

Логістична регресія застосовується для прогнозування вірогідності виникнення події за значеннями багатьох змінних. Для цього вводиться залежна змінна y, що приймає лише одне з двох значень - як правило, це цифри 0 (подія не відбувається) та 1 (подія відбувається), а також незалежних змінних (також називаються регрессорами), на основі значень яких потрібно знайти ймовірність прийняття того чи іншого значення залежної змінної [13].

Робиться припущення про те, що вірогідність події y = 1 дорівнює:

$$P\{y = 1|x\} = f(z),$$

де $z = \theta^T x = \theta_0 + \theta_1 x_1 + ... + \theta_n x_n$, x, θ – вектори-стовбці значень незалежної змінної і параметрів (коефіцієнтів регресії), а $f(z)$ – так звана логістична функція, сігмоїд або логіт-функція (рис. 2.3):

$$f(z) = \frac{1}{1+e^{-z}}.$$

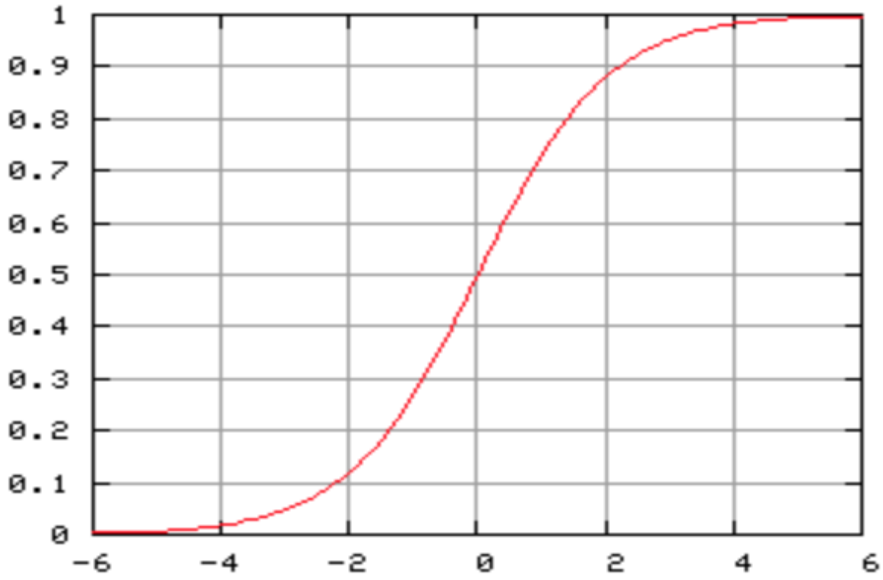

Рисунок 2.3 – Вигляд логістичної функції

Оскільки y приймає лише значення 0 та 1, то ймовірність прийняти значення 0 дорівнює:

$$P\{y = 0|x\} = 1 - f(z) = 1 - f(\theta^T x).$$

Функцію розподілу y при заданому x можна записати в такому вигляді:

$$P\{y|x\} = f(\theta^T x)^y (1 - f(\theta^T x))^{1-y}, y \in \{0,1\}.$$

Фактично це є розподілом Бернуллі з параметром $f(\theta^T x)$.

Для підбору параметрів необхідно створити тренувальну вибірку, що складається з наборів значень незалежних змінних та значень залежної змінної, що їм відповідає. Формально це множина пар $\left(x^{(1)}, y^{(1)}\right), \ldots \left(x^{(m)}, y^{(m)}\right)$, де $x^{(i)} \in R^n$ – вектор значень незалежних змінних, $y^{(i)} \in \{0,1\}$ – відповідне значення залежної змінної.

Зазвичай використовується ММП - метод максимальної правдоподібності, згідно якому вибираються параметри, що максимізують значення функції правдоподібності на тренувальній вибірці:

$$\hat{\theta} = argmax_\theta L(\theta) = argmax_\theta \prod_{i=1}^{m} P\{y = y^{(i)} | x = x^{(i)}\}.$$

Максимізація функцій правдоподібності еквівалентна максимізації її логарифму. Для максимізації цієї функції може бути використаний, наприклад, метод градієнтного спуску. На практиці також застосовують метод Ньютона та стохастичний градієнтний спуск.

У даній роботі для максимізації функції був використаний метод BFGS (алгоритм Бройде - Флетчера - Гольдфарба – Шанно). Це ітераційний метод чисельної оптимізації, названий на честь його дослідників. Він відноситься до класу так званих квазіньютоновскіх методів. На відміну від ньютонівських методів в квазіньютоновскіх не виразовується безпосередньо гессіан функції, тобто немає необхідності знаходити часткові похідні другого порядку. Замість цього гессіан обчислюється наближено, виходячи із зроблених до цього кроків. Це дозволяє значно підвищити швидкість розрухунку та зменшити об’єм використовуваної пам’яті.

Існує багато причин через які процедури розбиття активно використовуються при підготовці вибірок до моделювання методом логістичної регресії. Основна причина в тому, що взаємозв'язок між змінною та подією не завжди лінійній, а побудова логістичної регресії передбачає наявність саме такого зв’язку [14, 15].

2.5 Крос-валідація

Крос-валідація - це методика навчання та оцінки моделі, яка розбиває дані на кілька частин і навчає кілька алгоритмів на цих частинах. Крос-валідація - це метод формування навчальної і тестової частини для навчання аналітичної моделі в умовах недостатності вихідних даних або нерівномірного представлення класів.

Для успішного навчання аналітичної моделі необхідно, щоб класи були представлені в навчальній множині приблизно в однаковій пропорції. Однак якщо даних недостатньо або процедура розбиття при формуванні навчальної частини була проведена невдало, один з класів може виявитися домінуючим. Це може викликати «перекіс» у процесі навчання, і домінуючий клас буде розглядатися як найбільш імовірний. Метод крос-валідації дозволяє уникнути цього.

Один цикл крос-валідації включає розбиття набору даних на частини, потім побудова моделі на одній частині (званої тренувальним набором), і валідація моделі на іншій частині (званої тестовим набором). Крос-валідація важлива для захисту від помилок, нав'язаних даними, особливо коли отримання додаткових даних важко або неможливо.

Припустимо, у нас є модель з одним або декількома невідомими параметрами, і набір даних, на якому ця модель може бути оптимізована (тренувальний набір). Процес підгонки оптимізує параметри моделі і робить модель настільки підходящої під тренувальний набір, наскільки це можливо. Якщо ми тепер візьмемо незалежний зразок даних для валідації моделі з того ж джерела, звідки ми взяли тренувальний набір даних, зазвичай виявляється, що модель описує тестові дані гірше, ніж тренувальний набір. Це називається перенавчанням (overfitting), і особливо часто зустрічається в ситуаціях, коли розмір тренувального набору невеликий, або коли число параметрів в моделі

велике. Крос-валідація це спосіб оцінити модель на різних наборах і обрати такі параметри моделі, при яких результати на тестовому наборі даних будуть найкращими.

В основі методу лежить поділ вихідних даних на k приблизно рівних блоків, наприклад k = 10. Потім на k - 1, тобто на дев'яти блоках, проводиться навчання моделі, а 10-й блок використовується для тестування. Процедура повторюється k раз, при цьому на кожному проході для перевірки вибирається новий блок, а навчання проводиться на тих, що залишилися (рис. 2.4).

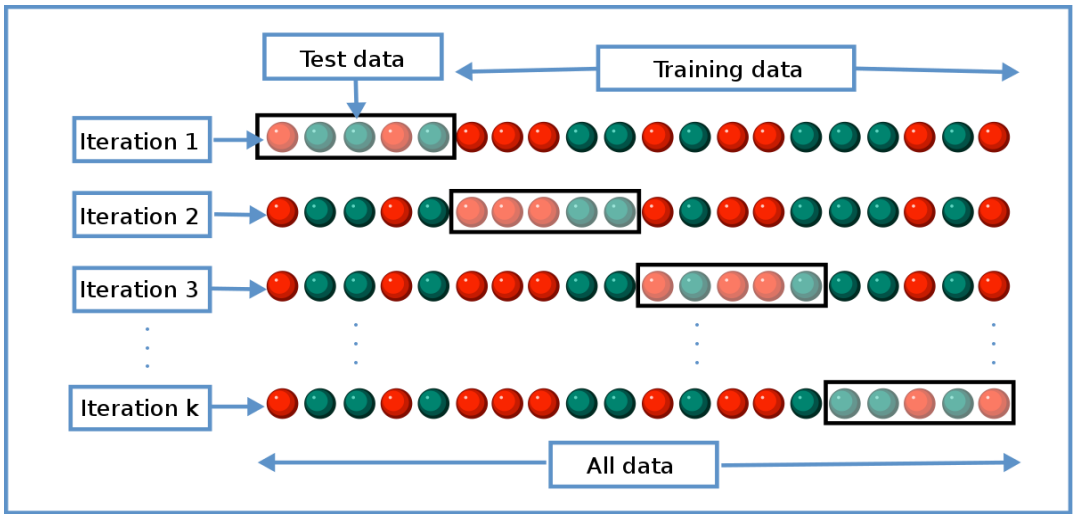

Рисунок 2.4 – Алгоритм крос-валідації

2.6 Тестування моделі

Оцінка якості передбачуваної можливості моделі - це один з найважливіших пунктів при побудові моделі. Важливо побудувати таку модель, яка буде добре визначати і «хороших» і «поганих».

Як правило, для оцінки якості моделі в машинному навчанні використовують ROC-криву, яка відображатиме співвідношення часток вірно знайдених ознак, що несуть результати (позитивних) і невірно знайдених, що

не несуть результатів (негативних). По-іншому ROC-крива називається кривою помилок, чим ближче крива до лівого верхнього кута графіка, тим краща передбачувана здатність моделі. Площа під кривою помилок - показник AUC - відображає якість класифікації моделі (рис. 2.5). Чим більше значення AUC, тим краще прогнозує модель [16].

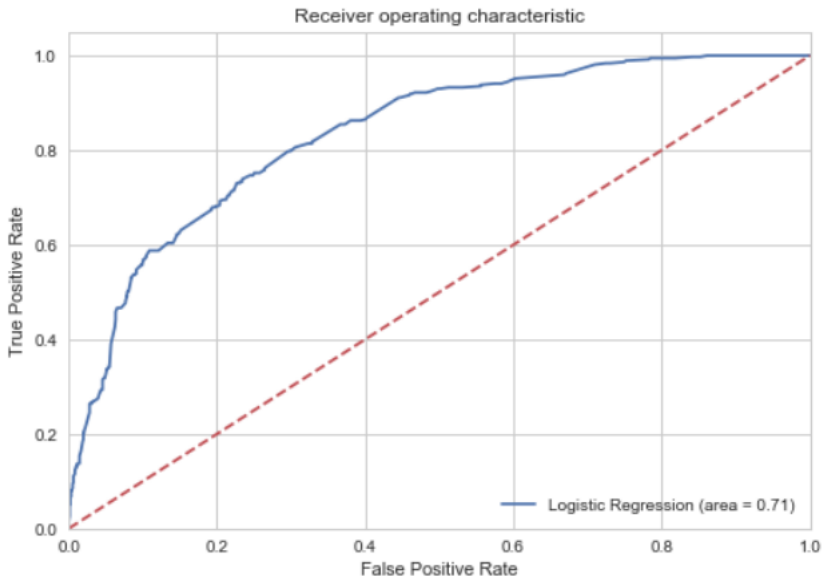

Рисунок 2.5 – Вигляд ROC-кривої

За значенням AUC обчислюється такий показник, як індекс Джині. Цей показник переводить значення площі під кривою в діапазон від 0 до 1 і обчислюється за формулою:

$$Gini = 2(AUC - 0.5).$$

Окрім прогностичної сили моделі необхідно перевірити, наскільки адекватно модель, побудована на історичних даних, буде працювати на новій історії. Саме тому необхідно оцінювати актуальність і репрезентативність навчальної вибірки відносно поточної вибірки, на якій модель використовується, щоб переконатися що вона відповідає даним.

Період щодо якого проводиться оцінка стабільності популяції називається базовим, а період для якого проводиться оцінка - цільовим. Для порівняння цих періодів використовується індекс стабільності популяції, або PSI. Індекс стабільності популяції є широко поширеною метрикою для моніторингу актуальності аналітичних моделей. Він відображає різницю між навчальною вибіркою і даними, на яких модель використовується.

Ця метрика дозволяє виміряти те, наскільки змінна змінилася в розподілі між двома вибірками з плином часу. Вона широко використовується для моніторингу змін характеристик популяції, діагностики можливих проблем з ефективністю моделі і розраховується за такою формулою:

$$PSI = \sum\left((\%Actual - \%Expected) * ln\frac{\%Actual}{\%Expected}\right),$$

де спостереження кожної змінної розбивається на 10 рівних груп (децилів) і $\%Actual$ – процент спостережень в кожній групі на цільовій вибірці, $\%Expected$ – процент спостережень в кожній групі на базовій вибірці.

Індекс стабільності популяції інтерпретується наступним чином:

а) PSI менше 10% показує відсутність значущої зміни в поточній вибірці;

б) PSI в діапазоні від 10 до 25% свідчить про незначній зміні, які необхідно досліджувати;

в) PSI більше 25% говорить про значне зміщення популяції і потрібно перестроювання моделі.

2.7 Вигляд скорингової карти

Фінальним етапом розробки скорингової моделі є переведення коефіцієнтів логістичної регресії в скорингові бали. Якщо взяти оцінки коефіцієнтів логістичної регресії і помножити їх на значення незалежних змінних, то отримаємо підсумковий скоринговий бал в шкалі натуральних логарифмів:

$$\text{бал} = \widehat{b_1} x_1 + \cdots + \widehat{b_k} x_k,$$

де x_j – значення незалежної змінної, $\widehat{b_j}$ – оцінки коефіцієнтів логістичної регресії.

Для приведення скорингових балів в лінійну шкалу використовують прийом масштабування. Масштабування не змінює прогностичну здатність скорингової карти, а лише переводить скорингові бали в нову шкалу, зручну для використання. Скоринговий бал в лінійній шкалі є відношенням шансів «хороших» позичальників до «поганих».

За основу взято загальноприйнятий стандарт FICO. В першу чергу задається діапазон, наприклад, від 0 від 1000. Потім вводиться такий показник D, що характеризує подвоєння шансу стати «хорошим» позичальником, найчастіше D = 40. Тобто кожні 40 балів у спостережуваного позичальника подвоюються його шанси стати «хорошим» позичальником.

Для приведення коефіцієнта логістичної регресії в скоринговий бал в лінійній шкалі застосовують наступне перетворення:

$$\text{бал} = A + R\widehat{b_j},$$

де R – множник, A – зсув.

Множник вираховується за формулою:

$$R = \frac{D}{\ln(2)}.$$

Зміщення шукається за формулою:

$$A = B + R * ln(C),$$

де B - значення в балах, в якому шанс стати хорошим позичальником становить C:1. Згідно FICO рекомендується брати C = 72 і B = 660, як загальноприйнятий стандарт розрахунку скорингових балів.

Якщо рівняння логістичної регресії було побудоване за значеннями WOE, то формула розрахунку скорингового бала в лінійному масштабі буде наступна:

$$\text{бал} = -\left(WoE_j * b_i + \frac{b_0}{n}\right) * R + \frac{A}{n},$$

де WoE_j - значення WoE для кожної j-ої категорії згрупованої змінної, n - кількість незалежних змінних в рівнянні регресії, b_0 - константа, b_i - коефіцієнт регресії для і-ой змінної.

Рейтингова система базується на отриманій скоринговій карті. Загальноприйнятою градацією балів є наступний розподіл:

а) 690 - 850 балів - відмінний бал;

б) 650 - 690 балів - стандартний бал;

в) 600 - 650 балів - задовільний бал;

г) 500 - 600 балів - бал нижче середнього;

д) 300 - 500 балів - погана оцінка.

Клієнти з балами нижче 600 потребують подальшого уважного аналізу.

2.8 Висновки

В даному розділі було описано математичний апарат, що буде застосовуватися для побудови та тестування скорингової моделі. Основна ідея полягає в використанні логістичної регресії на трансформованих значеннях незалежних змінних. В якості трансформації було запропоновано використати метод монотонного оптимального розбиття, що базується на показнику WoE. Такий підхід дозволяє максимізувати кореляцію між отриманими групами та залежною змінною, зберігаючи при цьому лінійний взаємозв’язок.

РОЗДІЛ 3 АНАЛІЗ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

3.1 Розрахунок характеристик

В якості вхідних даних була обрана вибірка, що містить у собі історичні дані по транзакціям юридичних осіб за період в два роки. Початкові характеристики складалися з: дати, коли був відкритий рахунок клієнта, інформації по контрагентам та сфері активності клієнта, балансу та обсягу вхідних та вихідних транзакцій на кожен день, а також типів самих транзакцій.

Дуже важливим на даному етапі є визначення можливих змінних, що можуть бути здобутими на основі вхідних даних та максимально відображати нестандартну чи підозрілу поведінку клієнта.

На основі вхідних характеристик було пораховано фінальні змінні на кожен день існування рахунку клієнта, що в подальшому планувалися використовуватися в моделюванні:

а) скільки днів у клієнта існує рахунок;
б) скільки контрагентів знаходяться у чорному списку;
в) скільки контрагентів не має кредитної історії у банку;
г) відношення сумарного балансу до сумарних транзакцій за місяць;
д) кількість змін сфери діяльності;
е) частка готівкових транзакцій за місяць;
ж) індекс Херфіндаля для транзакцій, що розраховується за формулою:

$$HI = {s_1}^2 + {s_2}^2 + \cdots + {s_n}^2,$$

де s_i – доля транзакцій за певний фіксований день до всіх транзакцій за місяць.

Даний індекс показує чи існує концентрація в проведенні транзакцій у якийсь певний день у місяці і чим ближчим є до одиниці, тим більша концентрація;

з) частка транзакцій з невідомим призначенням за місяць;

и) частка транзакцій без ПДВ за місяць;

к) частка транзакцій заокруглених до цілого значення за місяць;

л) коефіцієнт варіації для транзакцій за місяць;

м) коефіцієнт варіації для балансів за місяць.

Коефіцієнти варіації були розраховані за формулою:

$$cv = \frac{s}{\bar{x}},$$

де s – стандартне відхилення, $\bar{x}$ - вибіркове середнє.

У якості залежної змінної виступала змінна, що приймала значення 1 – у випадку, якщо клієнт виявився шахраєм, та 0 - інакше.

3.2 Застосування алгоритму розбиття

Вибірку було розбито на тренувальну та тестову частини випадково у співвідношенні 80%/20%. На тренувальній вибірці до усіх розрахованих характеристик був застосований алгоритм монотонного оптимального розбиття та отримано результати у вигляді таблиць з границями отриманих інтервалів та підрахованими значеннями для кожного з них:

а) Count – загальна кількість спостережень, що потрапила в інтервал;

б) Count(%) – відсоток спостережень, що потрапила у даний інтервал до усіх спостережень (не перевищує 5%);

в) Non-Event – кількість не-подій;

г) Event – кількість подій;

д) Event rate – частка подій;

е) показник WoE;

ж) показник IV.

Окрім таблиць можемо бачити розраховані фінальні показник IV для незалежної змінної, розбитої на інтервали та графіки тренду по WoE, аби дослідити напрямок зв'язку незалежної змінної з прогнозованою змінною та перевірити монотонність розбиття (рис. 3.1 – 3.28).

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 20.50)	192	0.057762	141	51	0.265625	-1.17495	0.124002
1	[20.50, 22.50)	183	0.055054	149	34	0.185792	-0.714302	0.037118
2	[22.50, 25.50)	359	0.108002	319	40	0.111421	-0.115576	0.001511
3	[25.50, 33.50)	1067	0.320999	969	98	0.091846	0.0994095	0.003049
4	[33.50, 35.50)	226	0.067990	207	19	0.084071	0.196392	0.002424
5	[35.50, 41.50)	524	0.157641	482	42	0.080153	0.248387	0.008807
6	[41.50, 49.50)	387	0.116426	358	29	0.074935	0.321349	0.010574
7	[49.50, 55.50)	188	0.056558	175	13	0.069149	0.407949	0.007998
8	[55.50, inf)	198	0.059567	190	8	0.040404	0.975695	0.038631
9	Special	0	0.000000	0	0	0.000000	0	0.000000
10	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.234113

Рисунок 3.1 – Результати розбиття для змінної Age

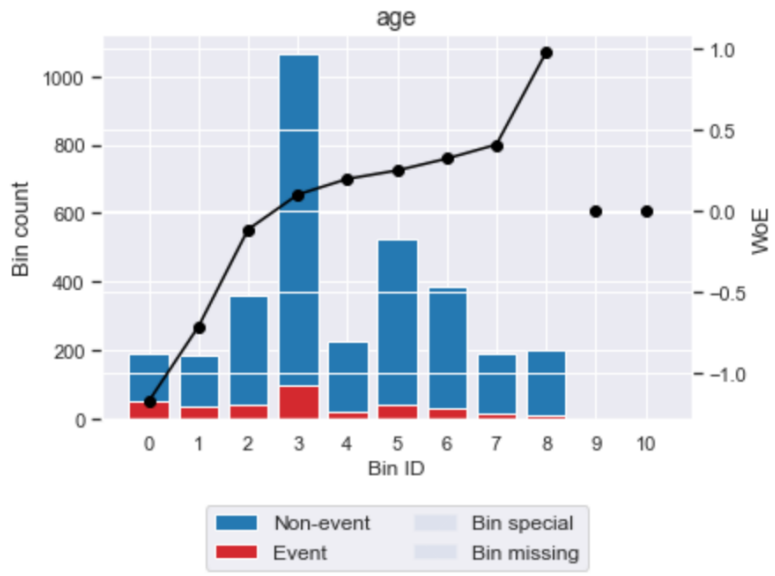

Рисунок 3.2 – Фінальні групи для змінної Age

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[False]	2856	0.859206	2590	266	0.093137	0.0840292	0.005866
1	[True]	468	0.140794	400	68	0.145299	-0.419931	0.029317
2	Special	0	0.000000	0	0	0.000000	0	0.000000
3	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.035183

Рисунок 3.3 – Результати розбиття для змінної Gender

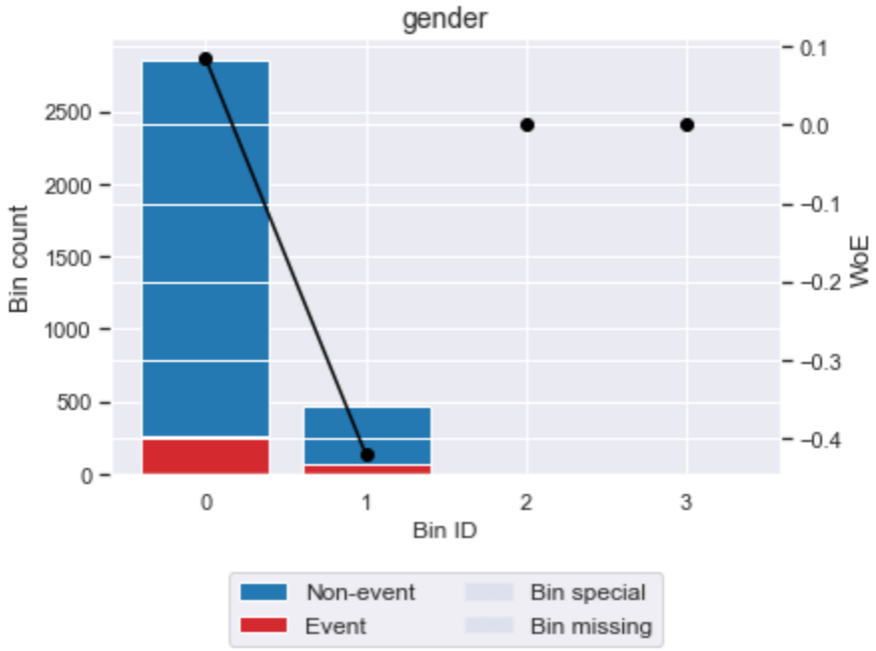

Рисунок 3.4 – Фінальні групи для змінної Gender

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 36.50)	281	0.084537	225	56	0.199288	-0.801139	0.074036
1	[36.50, 56.50)	243	0.073105	202	41	0.168724	-0.597192	0.032963
2	[56.50, 79.50)	294	0.088448	251	43	0.146259	-0.427635	0.019156
3	[79.50, 110.50)	356	0.107100	314	42	0.117978	-0.180164	0.003735
4	[110.50, 187.50)	742	0.223225	682	60	0.080863	0.238797	0.011570
5	[187.50, 251.50)	496	0.149218	459	37	0.074597	0.326245	0.013941
6	[251.50, 366.50)	529	0.159146	494	35	0.066163	0.4553	0.027512
7	[366.50, inf)	383	0.115223	363	20	0.052219	0.706783	0.043484
8	Special	0	0.000000	0	0	0.000000	0	0.000000
9	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.226399

Рисунок 3.5 – Результати розбиття для змінної Days_snc

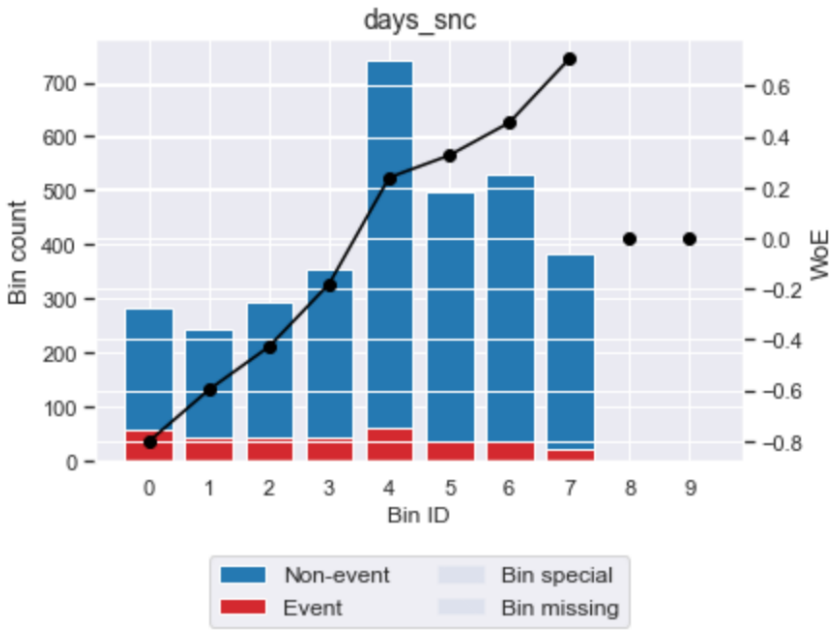

Рисунок 3.6 – Фінальні групи для змінної Days_snc

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 2.50)	2784	0.837545	2555	229	0.082256	0.220198	0.037188
1	[2.50, inf)	540	0.162455	435	105	0.194444	-0.770502	0.130127
2	Special	0	0.000000	0	0	0.000000	0	0.000000
3	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.167316

Рисунок 3.7 – Результати розбиття для змінної Num_cntrp_black

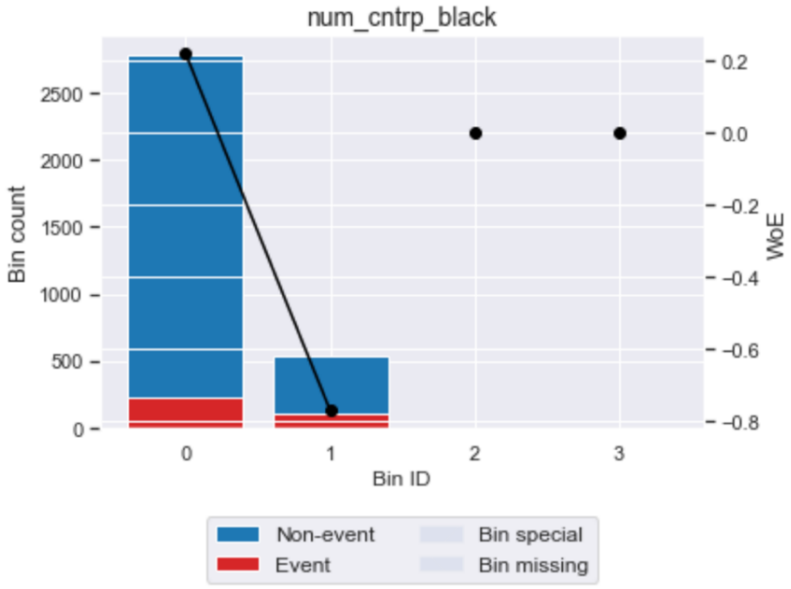

Рисунок 3.8 – Фінальні групи для змінної Num_cntrp_black

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 0.50)	3014	0.906739	2747	267	0.088587	0.139128	0.016602
1	[0.50, inf)	310	0.093261	243	67	0.216129	-0.903519	0.107815
2	Special	0	0.000000	0	0	0.000000	0	0.000000
3	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.124417

Рисунок 3.9 – Результати розбиття для змінної Num_cntrp_no_hist

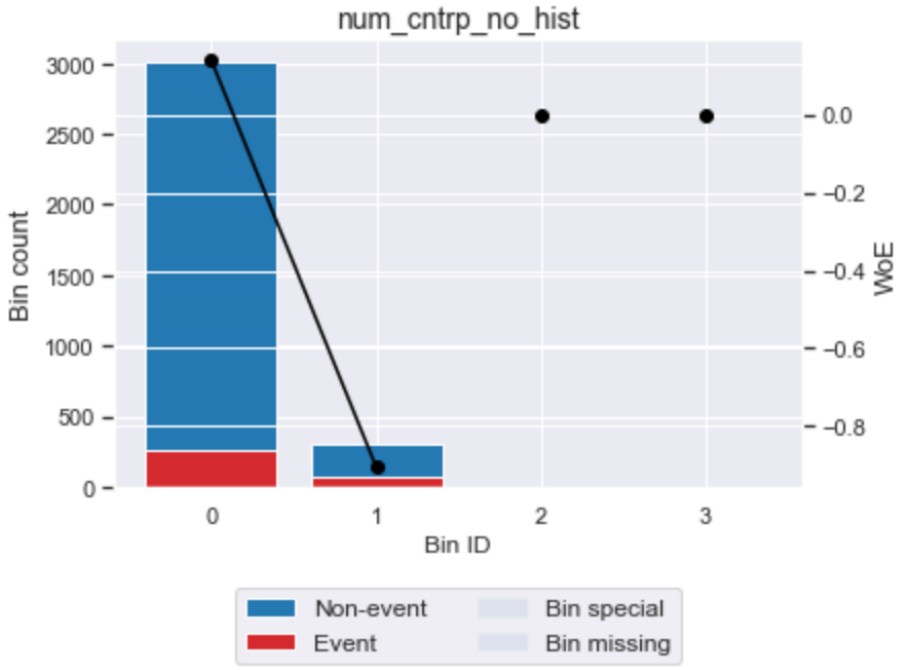

Рисунок 3.10 – Фінальні групи для змінної Num_cntrp_no_hist

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 1.13)	167	0.050241	135	32	0.191617	-0.752349	0.038112
1	[1.13, 1.43)	422	0.126955	350	72	0.170616	-0.610621	0.060153
2	[1.43, 1.73)	417	0.125451	356	61	0.146283	-0.427831	0.027198
3	[1.73, 1.91)	245	0.073706	212	33	0.134694	-0.331809	0.009257
4	[1.91, 2.26)	390	0.117329	360	30	0.076923	0.293019	0.008961
5	[2.26, 2.91)	716	0.215403	665	51	0.071229	0.376074	0.026217
6	[2.91, inf)	967	0.290915	912	55	0.056877	0.616419	0.086512
7	Special	0	0.000000	0	0	0.000000	0	0.000000
8	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.256411

Рисунок 3.11 – Результати розбиття для змінної Balance_to_trans

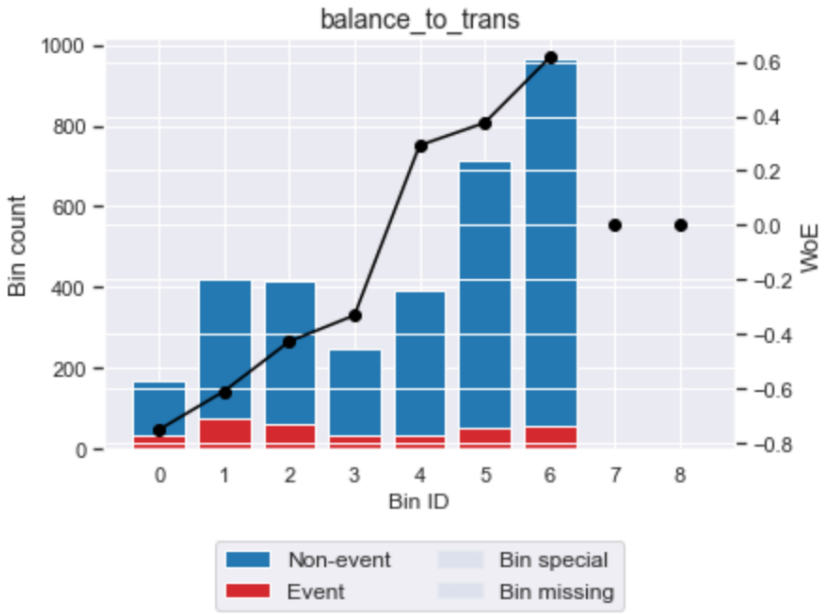

Рисунок 3.12 – Фінальні групи для змінної Balance_to_trans

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 2.50)	3093	0.930505	2787	306	0.098933	0.0172483	0.000275
1	[2.50, inf)	231	0.069495	203	28	0.121212	-0.210886	0.003361
2	Special	0	0.000000	0	0	0.000000	0	0.000000
3	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.003636

Рисунок 3.13 – Результати розбиття для змінної Num_act_ch

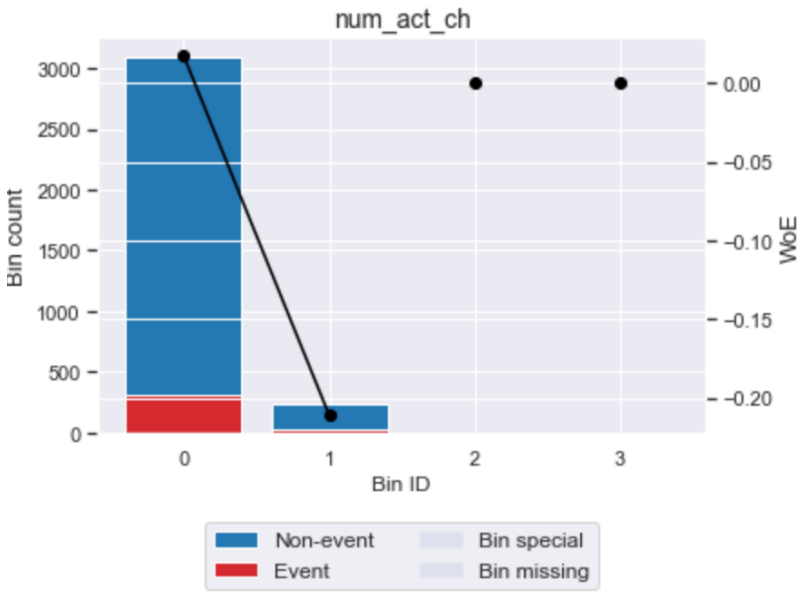

Рисунок 3.14 – Фінальні групи для змінної Num_act_ch

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 0.01)	173	0.052046	171	2	0.011561	2.25663	0.115545
1	[0.01, 0.01)	170	0.051143	155	15	0.088235	0.143487	0.000994
2	[0.01, 0.07)	1554	0.467509	1403	151	0.097169	0.0372006	0.000637
3	[0.07, 0.12)	1246	0.374850	1112	134	0.107544	-0.075812	0.002221
4	[0.12, inf)	181	0.054452	149	32	0.176796	-0.653677	0.030053
5	Special	0	0.000000	0	0	0.000000	0	0.000000
6	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.149451

Рисунок 3.15 – Результати розбиття для змінної Share_cash

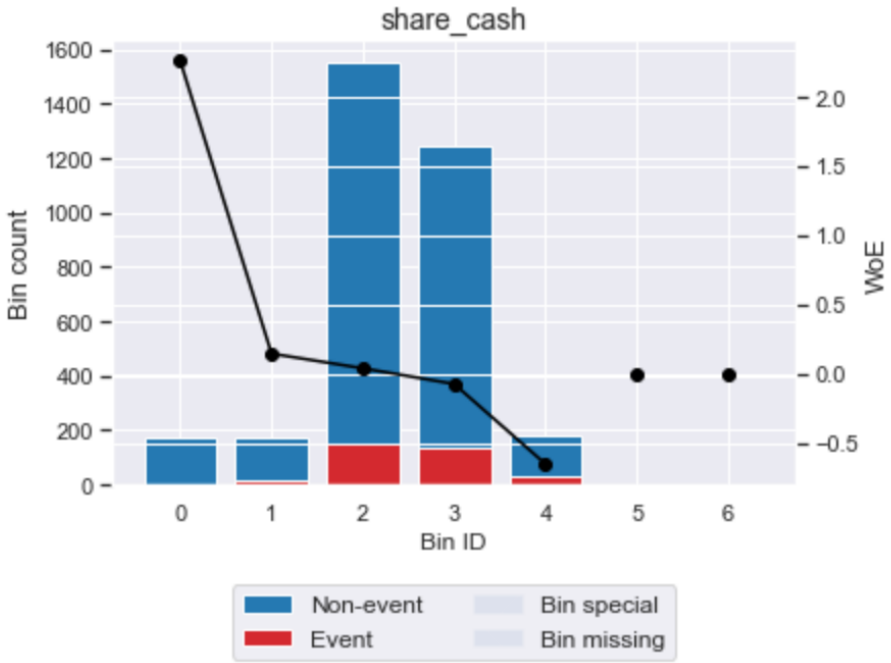

Рисунок 3.16 – Фінальні групи для змінної Share_cash

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 0.06)	171	0.051444	162	9	0.052632	0.698484	0.019023
1	[0.06, 0.20)	1019	0.306558	922	97	0.095191	0.0599466	0.001076
2	[0.20, 0.27)	652	0.196149	589	63	0.096626	0.0434038	0.000363
3	[0.27, 0.43)	1218	0.366426	1095	123	0.100985	-0.00556239	0.000011
4	[0.43, inf)	264	0.079422	222	42	0.159091	-0.52688	0.027135
5	Special	0	0.000000	0	0	0.000000	0	0.000000
6	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.047608

Рисунок 3.17 – Результати розбиття для змінної Herf_trans

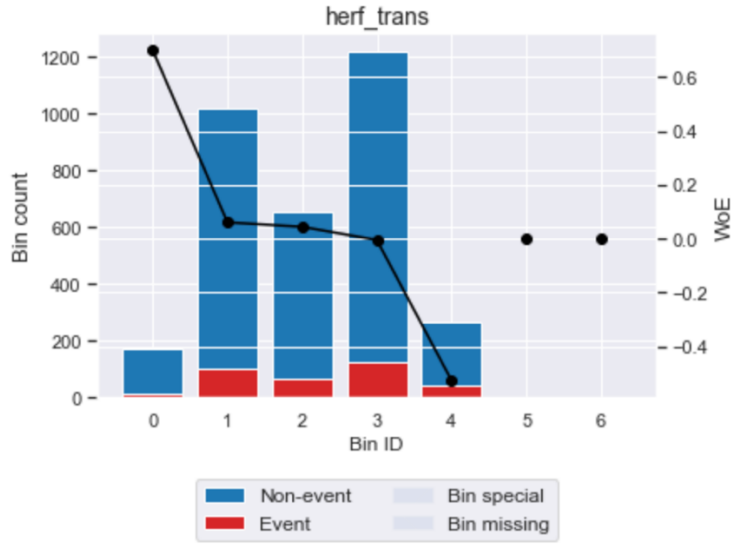

Рисунок 3.18 – Фінальні групи для змінної Herf_trans

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 0.06)	583	0.175391	552	31	0.053173	0.687673	0.063129
1	[0.06, 0.09)	311	0.093562	283	28	0.090032	0.121355	0.001313
2	[0.09, 0.24)	1793	0.539410	1618	175	0.097602	0.0322725	0.000555
3	[0.24, 0.26)	271	0.081528	239	32	0.118081	-0.18116	0.002876
4	[0.26, inf)	366	0.110108	298	68	0.185792	-0.714302	0.074235
5	Special	0	0.000000	0	0	0.000000	0	0.000000
6	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.142108

Рисунок 3.19 – Результати розбиття для змінної Share_unknwn_purp

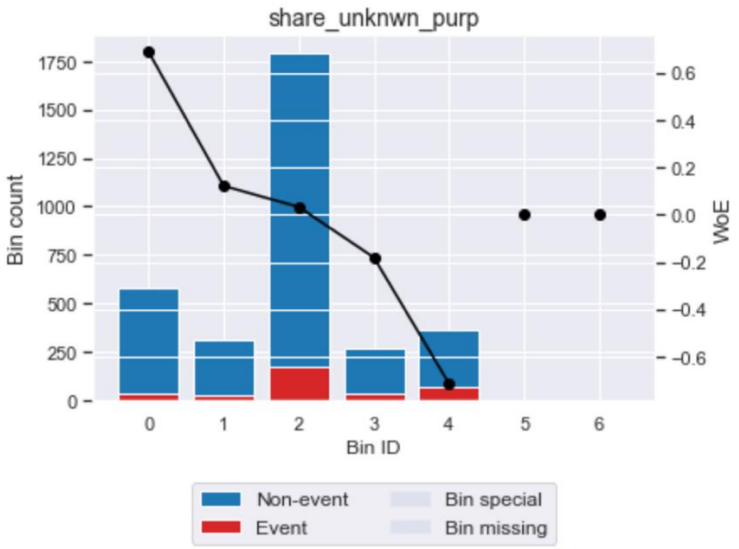

Рисунок 3.20 – Фінальні групи для змінної Share_unknwn_purp

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 0.15)	605	0.182010	585	20	0.033058	1.18399	0.160753
1	[0.15, 0.19)	173	0.052046	165	8	0.046243	0.834616	0.026067
2	[0.19, 0.59)	1868	0.561974	1676	192	0.102784	-0.0252178	0.000361
3	[0.59, 0.71)	511	0.153730	456	55	0.107632	-0.076728	0.000933
4	[0.71, inf)	167	0.050241	108	59	0.353293	-1.58729	0.223057
5	Special	0	0.000000	0	0	0.000000	0	0.000000
6	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.411170

Рисунок 3.21 – Результати розбиття для змінної Share_wo_VAT

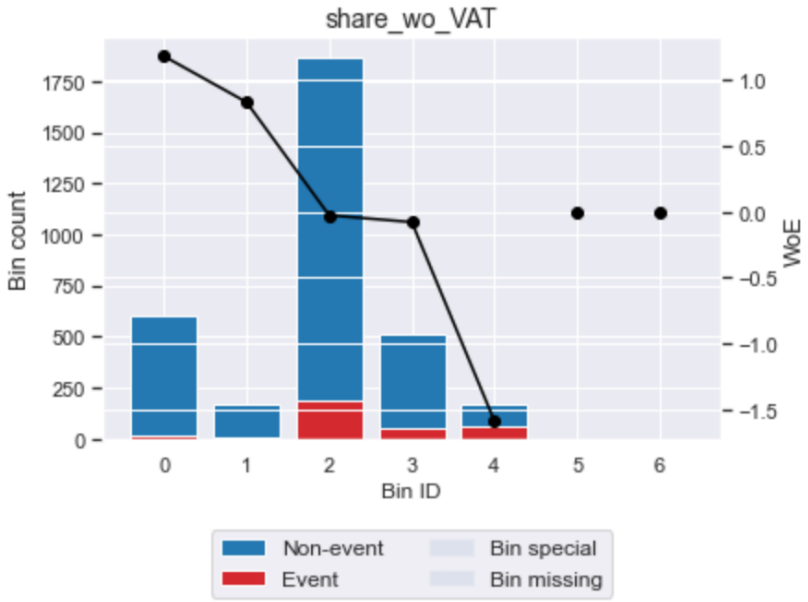

Рисунок 3.22 – Фінальні групи для змінної Share_wo_VAT

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 0.11)	197	0.059266	196	1	0.005076	3.08623	0.193068
1	[0.11, 0.14)	167	0.050241	164	3	0.017964	1.80937	0.082991
2	[0.14, 0.19)	269	0.080927	249	20	0.074349	0.329833	0.007717
3	[0.19, 0.26)	562	0.169073	514	48	0.085409	0.179135	0.005050
4	[0.26, 0.29)	273	0.082130	248	25	0.091575	0.102665	0.000831
5	[0.29, 0.35)	491	0.147714	445	46	0.093686	0.0775452	0.000861
6	[0.35, 0.37)	200	0.060168	177	23	0.115000	-0.151232	0.001462
7	[0.37, 0.47)	678	0.203971	589	89	0.131268	-0.302098	0.020989
8	[0.47, inf)	487	0.146510	408	79	0.162218	-0.550068	0.055046
9	Special	0	0.000000	0	0	0.000000	0	0.000000
10	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.368015

Рисунок 3.23 – Результати розбиття для змінної Share_round_trans

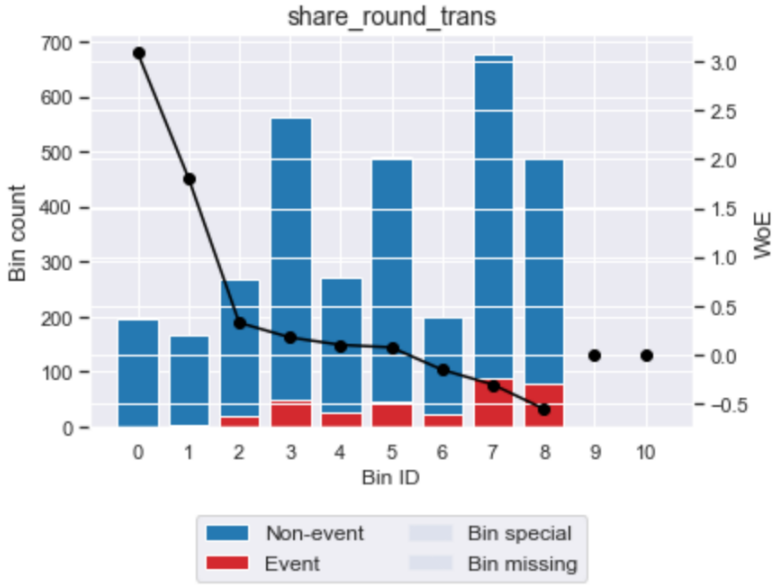

Рисунок 3.24 – Фінальні групи для змінної Share_round_trans

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 1.07)	180	0.054152	176	4	0.022222	1.5923	0.074658
1	[1.07, 1.34)	872	0.262335	794	78	0.089450	0.128487	0.004114
2	[1.34, 1.99)	2022	0.608303	1828	194	0.095945	0.0512319	0.001564
3	[1.99, inf)	250	0.075211	192	58	0.232000	-0.994835	0.108873
4	Special	0	0.000000	0	0	0.000000	0	0.000000
5	Missing	0	0.000000	0	0	0.000000	0	0.000000
Totals		3324	1.000000	2990	334	0.100481		0.189210

Рисунок 3.25 – Результати розбиття для змінної Cv_balance

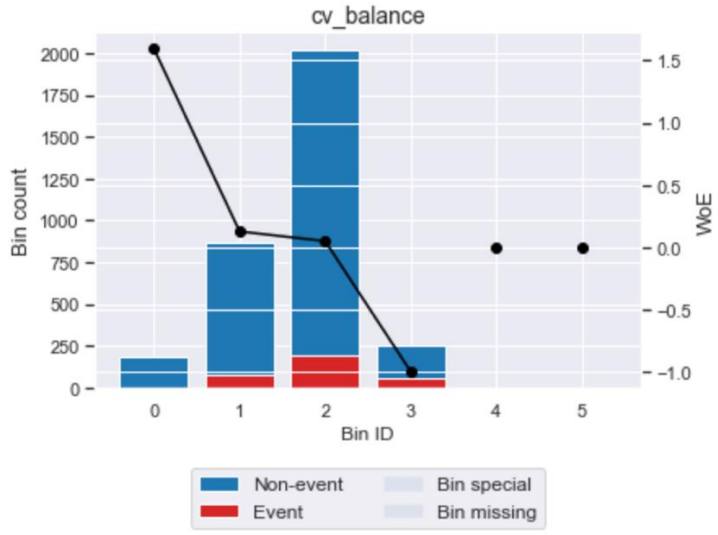

Рисунок 3.26 – Фінальні групи для змінної Cv_balance

	Bin	Count	Count (%)	Non-event	Event	Event rate	WoE	IV
0	[-inf, 1.66)	1127	0.339049	1058	69	0.061224	0.538141	7.924635e-02
1	[1.66, 1.88)	413	0.124248	374	39	0.094431	0.0688065	5.722735e-04
2	[1.88, 2.01)	239	0.071901	215	24	0.100418	0.000696524	3.487295e-08
3	[2.01, 2.90)	1369	0.411853	1218	151	0.110299	-0.104202	4.661787e-03
4	[2.90, inf)	176	0.052948	125	51	0.289773	-1.2954	1.436450e-01
5	Special	0	0.000000	0	0	0.000000	0	0.000000e+00
6	Missing	0	0.000000	0	0	0.000000	0	0.000000e+00
Totals		3324	1.000000	2990	334	0.100481		2.281255e-01

Рисунок 3.27 – Результати розбиття для змінної Cv_trans

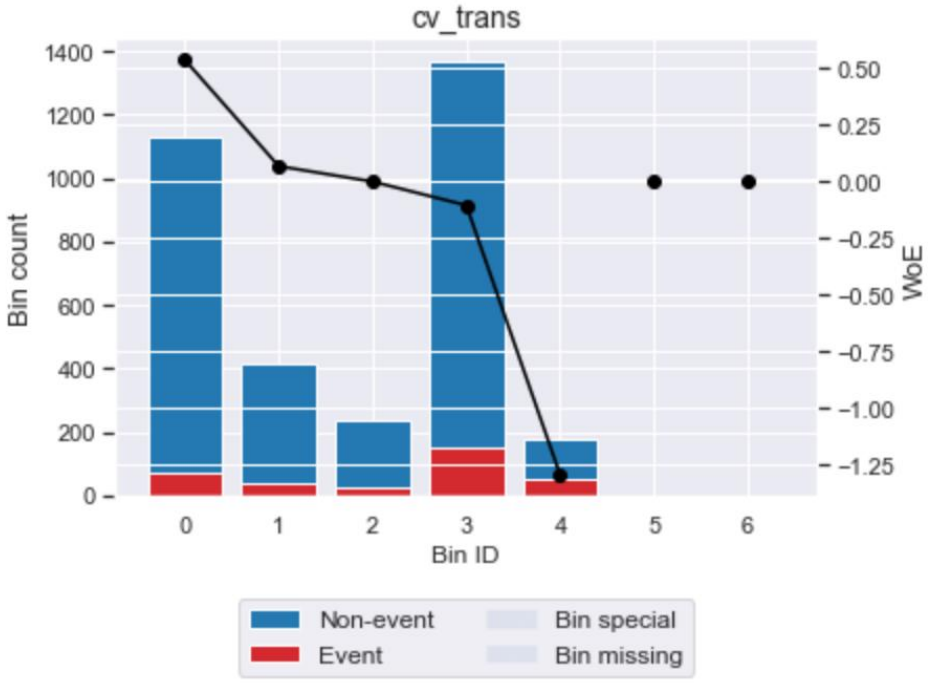

Рисунок 3.28 – Фінальні групи для змінної Cv_trans

Тренд усіх порахованих змінних є логічним та в правильному напрямку вказує на підозрілу бізнесову діяльність клієнта.

Можна виділити такий профіль поганого клієнта:

а) клієнт молодий;

б) досить нещодавно відкрив рахунок у банку;

в) має більше двох контрагентів у чорному списку;

г) має контрагентів без кредитної історії у банку;

д) має незначний баланс в порівнянні з проведеними транзакціями;

е) змінював сферу діяльності більше ніж два рази;

ж) має значну частку готівкових транзакцій;

з) має транзакції, що концентруються в конкретний день;

и) має значну частку транзакцій з невідомим призначенням;

к) має значну частку транзакцій без ПДВ;

л) має значну частку заокруглених до цілого значення транзакцій;

м) має нестабільний у часі баланс;

н) має нестабільні у часі транзакції.

Серед обраних змінних варто залишити лише ті, що мають досить сильну предикативну можливість. Для цього скористаємося показником IV (табл. 3.1).

Таблиця 3.1 – Інтерпретація значень IV

IV	Предикативна сила
< 0.02	недоцільно для прогнозування
0.02 - 0.1	слабке прогнозування
0.1 - 0.3	середнє прогнозування
0.3 - 0.5	сильне прогнозування
> 0.5	дуже сильне прогнозування, майже нереальне

Відкинемо перед моделюванням змінні, що мають показник менший за 0.02. В нашому випадку це лише одна змінна – кількість змін сфер діяльності.

Подальший аналіз буде відбуватися не з початковими значеннями характеристик, а з відповідно присвоєними значеннями WoE.

3.3 Побудова і тестування логістичної регресії

Як відомо, мультиколінеарність - це ситуація, в якій незалежні змінні регресії мають зв'язки між собою. Даний факт може зменшити стійкість оцінок параметрів регресії та збільшити дисперсії цих оцінок. Для того аби уникнути цього, було розглянуто коефіцієнт рангової кореляції Спірмена для кожної з пар трансформованих змінних (рис. 3.29).

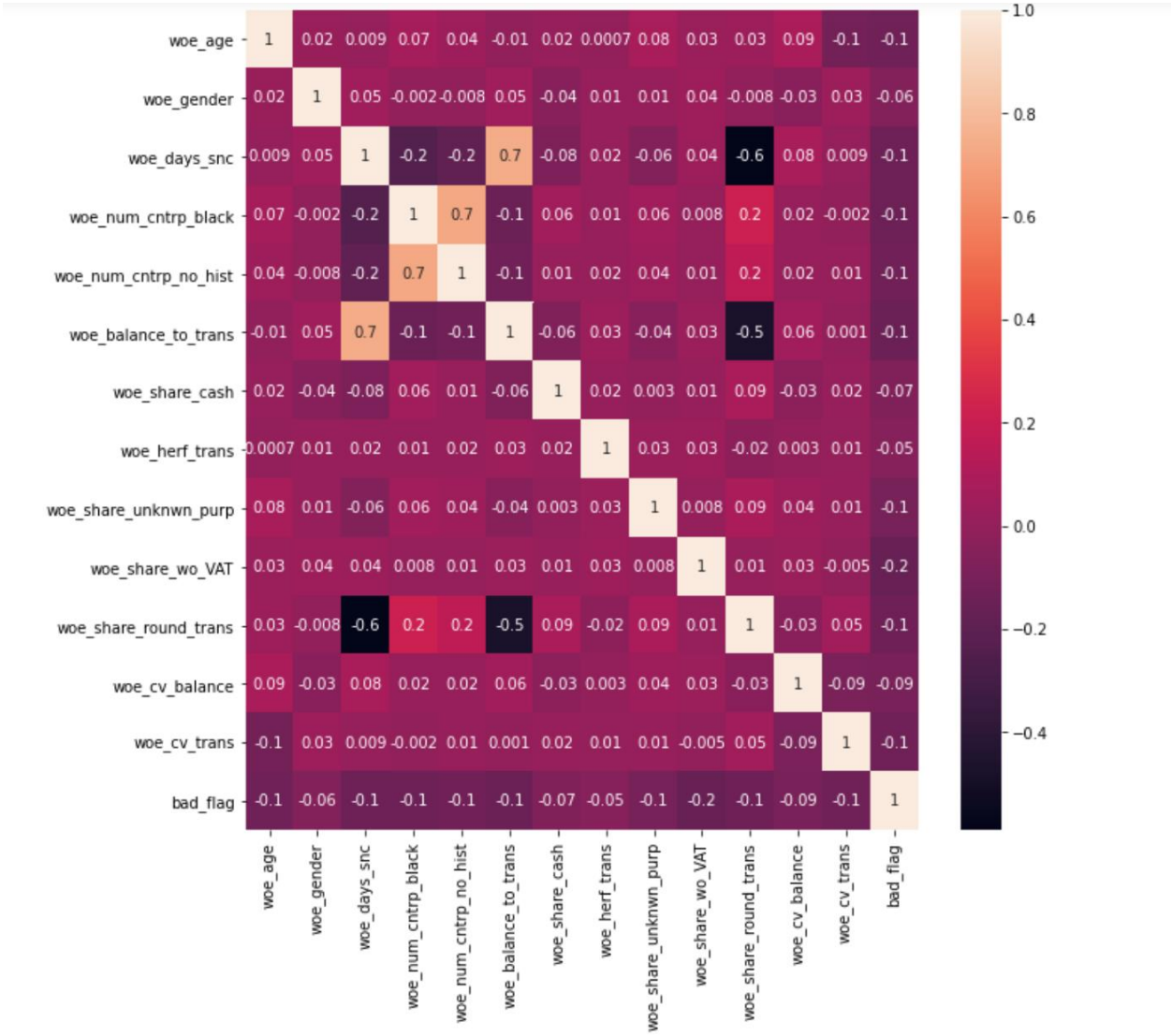

Рисунок 3.29 – Кореляційна матриця для трансформованих змінних

Для оцінки сили зв’язку між ними була використана шкала Чеддока (табл. 3.2).

Таблиця 3.2 – Шкала Чеддока

Значення кореляції	Інтерпретація
0 – 0.3	дуже слабка
0.3 – 0.5	слабка
0.5 – 0.7	середня
0.7 – 0.9	висока
0.9 - 1	дуже висока

Як видно, ні для одної пари не спостерігається сильний зв’язок, тому усі змінні після цього кроку залишаються для подальшого моделювання.

Була побудована логістична регресія на 13 вхідних змінних, де їх значення були замінені відповідним значенням WoE, в залежності від інтервалу, в який це значення потрапляло. Отримані коефіцієнти були переведені в шкалу скорів. На даному етапі важливо ще раз перевірити, що тренд характеристик зберігся, тобто зі зростанням значень змінної відповідно збільшуються (або зменшуються) скори. Клієнти були проскорені відповідно до відповідних значень характеристик, отриманий на тренувальній вибірці розподіл скорів показаний на рис. 3.30.

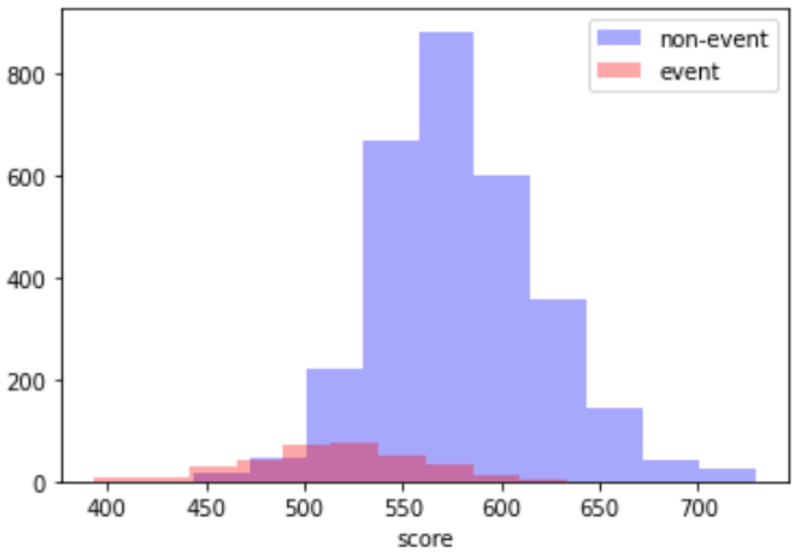

Рисунок 3.30 – Розподіл скорів на тренувальній вибірці

Також за допомогою описаних в попередньому розділі тестів була перевірена прогностична сила побудованої моделі на тренувальній (рис. 3.31) та тестовій вибірках (рис. 3.32).

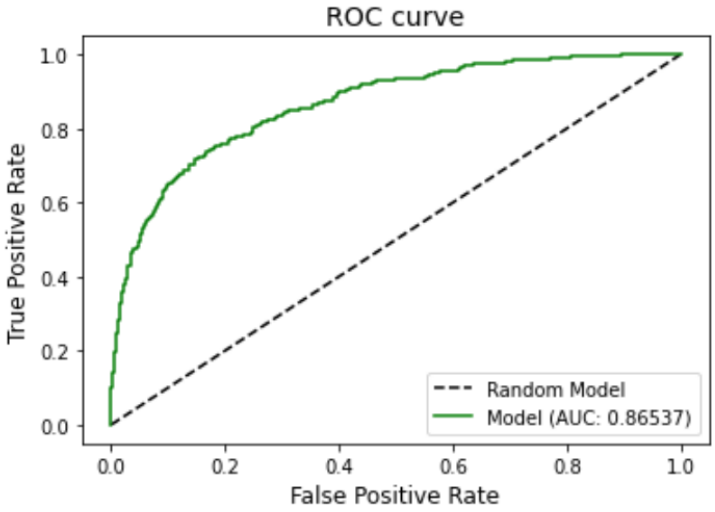

Рисунок 3.31 – ROC крива для тренувальної вибірки

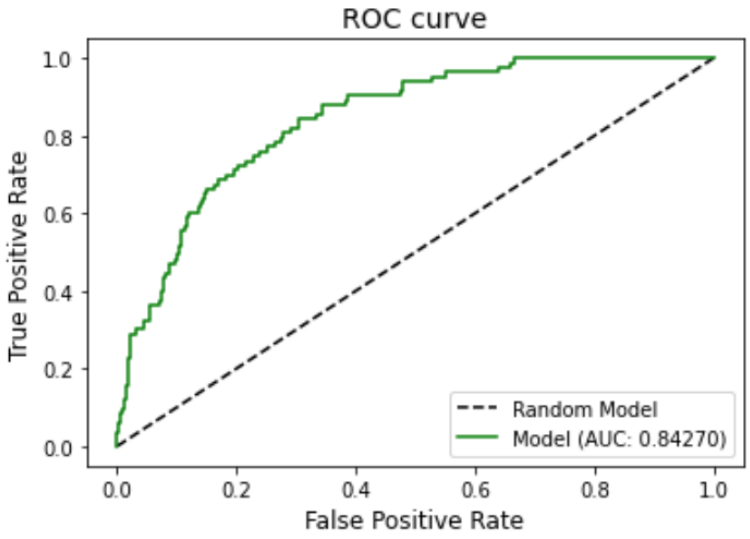

Рисунок 3.32 – ROC крива для тестової вибірки

За табл. 3.3 значень AUC можна зробити висновок, що модель дуже добре прогнозує як на тренувальній так і на тестовій вибірках.

Таблиця 3.3 – Інтерпретація значень AUC

AUC	Якість
0.9 - 1	Відмінна
0.8 – 0.9	Дуже добра
0.7 – 0.8	Добра
0.6 – 0.7	Середня
0.5 – 0.6	Незадовільна

Також був розрахований коефіцієнт Gini:

а) Gini на тренувальній вибірці – 73.07%;

б) Gini на тестовій вибірці – 68.54%.

Природньо, що на тестовій вибірці модель працює гірше, але при цьому вона залишається дуже сильною.

3.4 Пошук найкращої моделі за допомогою крос-валідації

Для підбору найкращих параметрів моделі було зроблено спробу побудовати логістичну регресію з тими самими змінними, але з використанням крос-валідаціїї з розбиттям на 10 частин. Було обрано кращу модель за результатами на тестових частинах, аналогічно до попередньої моделі перевірено тренд характеристик за скорами, результати показано на рис. 3.33- 3.35.

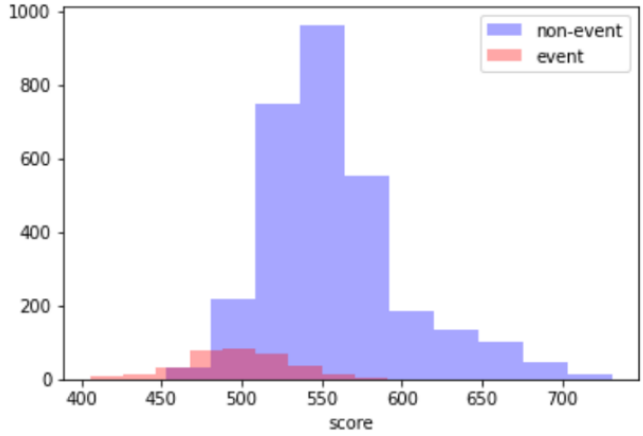

Рисунок 3.33 – Розподіл скорів на тренувальній вибірці

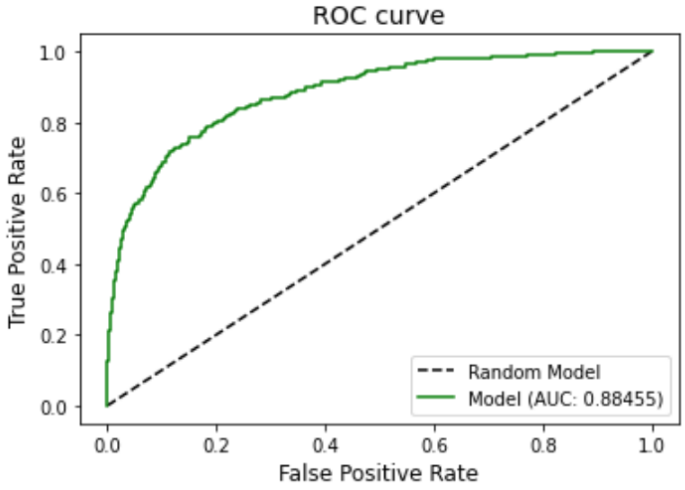

Рисунок 3.34 – ROC крива для тренувальної вибірки

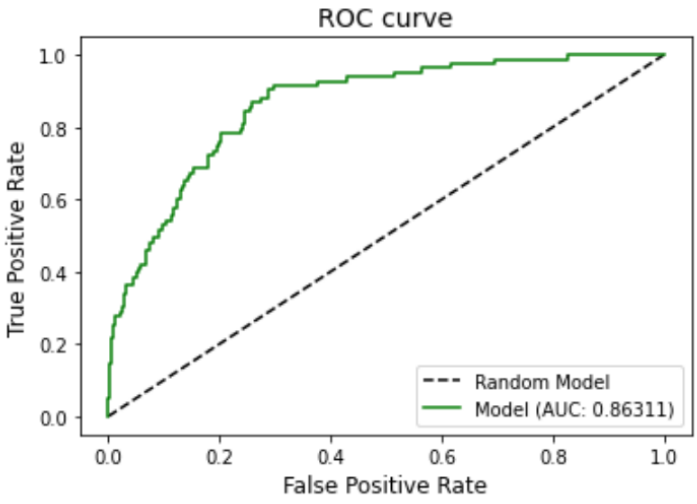

Рисунок 3.35 – ROC крива для тестової вибірки

Також був розрахований коефіцієнт Gini:

а) Gini на тренувальній вибірці – 76.91%;

б) Gini на тестовій вибірці – 72.62%.

З результатів видно, що модель показує кращі результати ніж попередня, а, отже, ця модель буде обрана як фінальна.

Для перевірки стабільності популяції було розраховано індекс стабільності популяції, де тренувальна вибірка порівнювалася з даними за останній квартал. Результати для кожної змінної можна бачити на рис. 3.36.

	Variable	PSI
0	age	0.009524
1	balance_to_trans	0.002106
2	cv_balance	0.004864
3	cv_trans	0.005626
4	days_snc	0.006078
5	gender	0.003537
6	herf_trans	0.016642
7	num_cntrp_black	0.000460
8	num_cntrp_no_hist	0.000232
9	share_cash	0.007945
10	share_round_trans	0.004869
11	share_unknwn_purp	0.007757
12	share_wo_VAT	0.005587

Рисунок 3.36 – Показник PSI для фінальних змінних

Як видно, усі змінні мають значення показника менше 10%, а отже їх розподіл є стабільним в порівнянні з базовою вибіркою.

Також стабільність була перевірена для фінальних скорів (рис. 3.37).

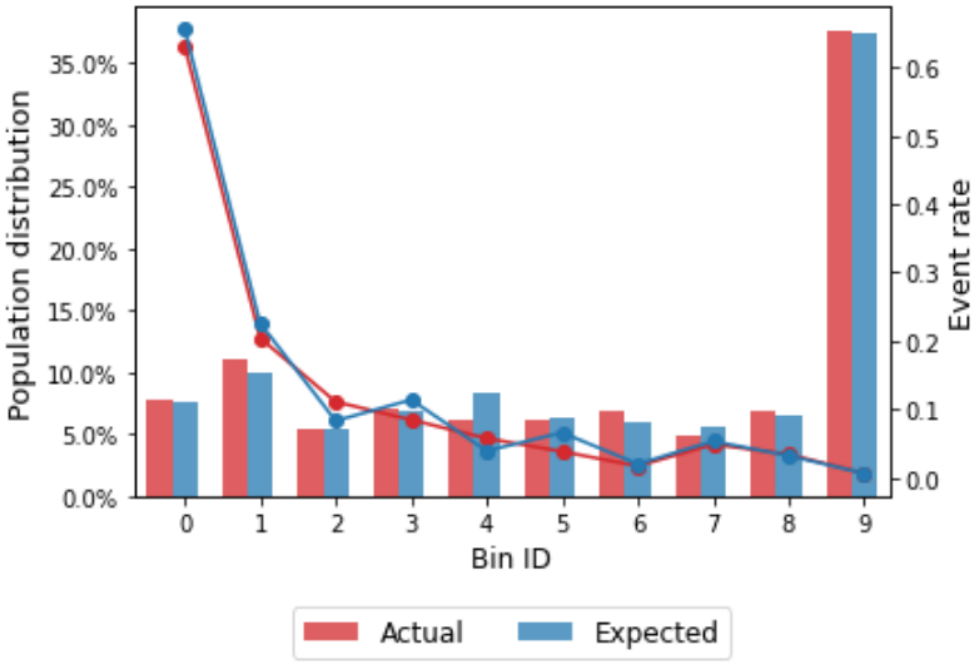

Рисунок 3.37 – Розподіл скорів тренувальної вибірки в порівнянні з базовою

Значення PSI = 1.08%, отже робимо висновок, що вибірка репрезентативна.

3.5 Висновки

В даному розділі було описано процес аналізу вхідних даних та розрахунок фінальних характеристик, що можуть відображати підозрілу поведінку клієнта. На цих характеристиках був застосований алгоритм монотонного оптимального розбиття, після чого змінні зі слабкою прогнозованою силою були виключені з моделювання. Було досліджено кореляцію між отриманими групами змінних та побудовано модель логістичної регресії, коефіцієнти якої були переведені в шкалу скорів. Модель була протестована і показала хороші результати як на тренувальній так і на тестовій вибірках. Був реалізований пошук найкращих параметрів за допомогою крос-валідації і в результаті була отримана модель з кращими

результатами, яка і була обрана фінальною. Розподіл тренувальної вибірки цієї моделі був порівняний з розподілом базової вибірки, з чого був зроблений висновок, що вибірка, яка використовувалася для моделювання, репрезентативна. Отже, побудована фінальна модель дає дуже хороші результати і може бути застосована для скорингу клієнтів.

РОЗДІЛ 4 СТАРТАП ПРОЕКТ «АНТИФРОД»

Основна ідея магістерської дисертації полягає в тому, аби створити автоматизовану систему запобігання шахрайства. Такий програмний продукт можна продати фінансовим установам, що досі використовують ручний підхід для таких цілей.

4.1 Опис ідеї проекту

У табл. 4.1 надано зміст ідеї, можливі напрямки застосування та основні вигоди, що може отримати користувач товару.

Таблиця 4.1 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Створення програмного продукту «Антифрод»	1. Звуження сірої зони клієнтів для перевірки верифікаторами	Економія ресурсів
	2. Зменшення ризиків видачі кредитів шахраям	Зменшення ризиковості кредитного портфелю
	3. Аналіз бізнесу та можливостей потенційних клієнтів	Можливість пропонувати нові продукти

Виділимо такі техніко-економічні характеристики ідеї:

а) точність оцінювання ризику;

б) аналіз клієнта з досить різних боків;

в) гнучкість та зручність у використанні.

Для порівняння цього продукту з іншими представниками на ринку, у якості конкурентів виберемо такі три програмні продукти:

а) Фрод-аналіз;

б) Фродекс;

в) ФродВол.

Таблиця 4.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Анти фрод	Фрод-аналіз	Фродекс	Фрод Вол			
1.	оцінювання ризику	+	-	+	+			+
2.	аналіз	-	+	-	+	+		
3.	гнучкість та зручність	+	+	-	+		+	

4.2.Технологічний аудит ідеї проекту

Визначення технологічної здійсненності ідеї проекту передбачає аналіз складових, наведених у табл. 4.3.

Таблиця 4.3 – Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Розрахунок нових характеристик	Купівля даних у фінансових установах	Наявні	+
2	Застосування нових методів	Користування послугами аналітиків	Наявні	+
3	Покращення інтерфейсу	Користування послугами програмістів та дизайнерів	Наявні	+
Обрана технологія реалізації ідеї проекту: залучення інвесторів задля вкладання коштів у купівлю нових даних, покращення існуючих характеристик, методів та інтерфейсу				

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Спочатку проводиться аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (табл. 4.4).

Таблиця 4.4 Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	5
2	Загальний обсяг продаж, грн	3 млн
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу	Прозорість аналізу

Продовження таблиці 4.4

5	Специфічні вимоги до стандартизації та сертифікації	-
6	Середня норма рентабельності в галузі (або по ринку), %	30

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (табл. 4.5).

Таблиця 4.5 – Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Потреба у запобіганні шахрайству	Банки та інші фінансові установи України	-	- ефективність; - прозорість; - зручність.

Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (табл. 4.6 і 4.7). Фактори в таблиці подані в порядку зменшення значущості.

Таблиця 4.6 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Помилки в даних	При помилках в даних прогнозування може дати невірні результати	Виявлення причини помилки та повідомлення фінансовій установі
2	Людська помилка при роботі з продуктом	Ненавмисна неправильна дія під час запуску програмного продукту	Штрафування працівника, що допустився помилки

Таблиця 4.7 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Ініціативність та креативність працівників	Розробка нових методів та аналізів	Заохочення таких працівників бонусами та преміями
2	Співпраця з великою кількістю установ	Обробка та аналіз великих масивів даних	Дії напрямлені на продовження такої співпраці

Далі проводиться аналіз пропозиції: визначаються загальні риси конкуренції на ринку (табл. 4.8).

Таблиця 4.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія / олігополія / монополістична / чиста	Чиста	Гарні перспективи розвитку
2. За рівнем конкурентної боротьби - локальний / національний / …	Національний	Ведучи конкуренцію на національному рівні, компанії необхідно прикласти належні зусилля для охоплення всього національного ринку.
3. За галузевою ознакою - міжгалузева / внутрішньогалузева	Внутрішньогалузева	Необхідно зосередити зусилля на пошуку конкурентних переваг, які дозволять компанії займати стійкі конкурентні позиції в даній галузі.
4. Конкуренція за видами товарів: - товарно-родова - товарно-видова - між бажаннями	Товарно-видова	Необхідно зосередитися на перевагах даного програмного продукту серед інших існуючих.

Продовження таблиці 4.8

5. За характером конкурентних переваг - цінова / нецінова	Нецінова	Зосередити зусилля на точності прогнозування та надійності продукту
6. За інтенсивністю - марочна / не марочна	Марочна	Дослідити якість послуг та цінову категорію конкурентів

Після аналізу конкуренції проводиться більш детальний аналіз умов конкуренції в галузі (табл. 4.9).

Таблиця 4.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Навести перелік прямих конкурентів	Визначити бар’єри входження в ринок	Визначити фактори сили постачальників	Визначити фактори сили споживачів	Фактори загроз з боку замінників
Висновки	На ринку спостерігається збільшення кількості гравців, що сприяє кращій конкуренції.	Обов’язковою є прозорість розробки продукту.	Немає залежності від постачальників.	Клієнти значною мірою впливають на загальний попит.	Субститутів немає.

За результатами аналізу таблиці робиться висновок щодо принципової можливості роботи на ринку з огляду на конкурентну ситуацію (табл. 4.10).

Таблиця 4.10 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Прозорість підходу	Залежності і методи чітко проглядаються та аналізуються
2	Гарантії ефективної роботи	Висока точність прогнозів дозволяє краще попереджати шахрайство
3	Зручність у використанні	Легка інтерпретуємість та гнучкість

За визначеними факторами конкурентоспроможності (табл. 4.10) проводиться аналіз сильних та слабких сторін стартап-проекту (табл. 4.11).

Таблиця 4.11 – Порівняльний аналіз сильних та слабких сторін «Смарт Дебт»

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з «Антифрод»						
			–3	–2	–1	0	+1	+2	+3
1	Прозорість підходу	20						+	
2	Гарантії ефективної роботи	18				+			
3	Зручність у використанні	15			+				

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (табл. 4.12).

Таблиця 4.12 – SWOT- аналіз стартап-проекту

Сильні сторони: прозорість підходу, гарантії ефективної роботи, зручність у використанні	Слабкі сторони: відсутність великої кількості даних, відсутність великої кількості досвідченого персоналу
Можливості: ініціативність та креативність працівників, співпраця з великою кількістю установ	Загрози: помилки в даних, людська помилка при роботі з продуктом

На основі SWOT-аналізу розробляються альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок (див. табл. 9, аналіз потенційних конкурентів).

Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів (табл. 4.13).

Таблиця 4.13 – Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Залучення інвестицій	Пошук інвесторів для залучення коштів задля співпраці з розробниками та аналітиками	3 місяці
2	Розвиток ринку	Вихід продукту на міжнародний ринок	12 місяців

4.4 Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (табл. 4.14).

Таблиця 4.14 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Важкість входу у сегмент
1	Банки	середня	середній	висока	висока

Продовження таблиці 4.14

2	Експрес-кредитування	висока	середній	висока	середня
Які цільові групи обрано: компанії експрес-кредитування.					

Для роботи в обраних сегментах ринку необхідно сформувати базову стратегію розвитку (табл. 4.15).

Таблиця 4.15 – Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Спеціалізація	Зосередження на одному сегменті ринку	гарантії ефективної роботи, зручність у використанні	Стратегія спеціалізації

Наступним кроком є вибір стратегії конкурентної поведінки (табл. 4.16).

Таблиця 4.16 – Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Забиратиме існуючих у конкурентів	Напрямок у сторону виділення переваг на фоні конкурентів	Стратегія заняття конкурентної ніші

На основі вимог споживачів з обраних сегментів до продукту (див. табл. 4.5), а також в залежності від обраної базової стратегії розвитку (табл. 4.15) та стратегії конкурентної поведінки (табл. 4.16) розробляється стратегія позиціонування (табл. 4.17). що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати проект.

Таблиця 4.17 – Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	ефективність, зручність, гнучкість.	Стратегія спеціалізації	гарантії ефективної роботи, зручність у використанні	ефективність, зручність, гнучкість

4.5 Розроблення маркетингової програми стартап-проекту

Першим кроком є формування маркетингової концепції товару, який отримає споживач. Для цього у табл. 4.18 потрібно підсумувати результати попереднього аналізу конкурентоспроможності товару.

Таблиця 4.18. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Ефективність прогнозування	Точність оцінки та прогнозу забезпечує високу ефективність	Більшість конкурентів також за цим слідкують

Продовження таблиці 4.18

2	Зручність у використанні програмного продукту	Зрозумілість програмного продукту та легкість у внесенні змін	Більшість конкурентів використовують досить закриті системи

Надалі розробляється трирівнева маркетингова модель товару: уточнюється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання (табл. 4.19).

Таблиця 4.19 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Розумний програмний продукт із високим рівнем точності оцінювання та покроковим аналізом.		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. Функціональність 2. Зручність 3. Зовнішній вигляд	М М Нм	Тх Е Е/Ор
	Якість: продукт має відповідати міжнародним стандартам		
III. Товар із підкріпленням	Використовуються тимчасові знижки		
За рахунок чого потенційний товар буде захищено від копіювання: за рахунок інтелектуальної власності.			

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту), яке передбачає аналіз ціни на товари-аналоги або товари

субститути, а також аналіз рівня доходів цільової групи споживачів (табл. 4.20).

Таблиця 4.20 – Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	Немає	Трохи вищий	Високий	від 10% до 40% від зборів

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення (табл. 4.21).

Таблиця 4.21 – Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Банки надають у роботу дані по заявкам на кредити	Робити звіти про результати виконаної роботи	Без посередників	Напряму з клієнтом

Останньою складової маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (табл. 4.22).

Таблиця 4.22 – Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Дані не мають потрапити у сторонні руки	Телефон, e-mail	Забезпечення якості та ефективності	Привернути увагу та зацікавити цільових клієнтів.	Надання знижок на перші місяці користування продуктом.

4.6 Висновки

У даному розділі було розглянуто спроможності розробленого програмного продукту для запобігання кредитного шахрайства. Було проаналізовано ринок та конкурентів, описані основні стратегії та можливості. Даний продукт доцільно розвивати та реалізовувати в сучасних умовах.

ВИСНОВКИ

У даній роботі було проведено аналіз даних по транзакціям юридичних осіб за період в два роки. На основі цих даних було створено та розраховано характеристики, які можуть відображати підозрілі дії клієнта у користуванні карткою чи у веденні бізнесу.

Для підвищення стабільності моделі та збереження лінійного зв'язку з прогнозованою змінною розраховані характеристики були розбиті на інтервали методом оптимального монотонного розбиття.

Для кожної змінної був отриманий логічний тренд та досліджена сила зв'язку із залежною змінною за допомогою показника IV, за результатами якого слабкі змінні були виключені з подальшого моделювання.

На відповідних для кожного інтервалу незалежної змінної показниках WoE була побудована модель логістичної регресії. Тестування проводилося за допомогою ROC-кривої та дало хороші результати як на тренувальній так і на тестовій вибірках. Для покращення цих результатів була зроблена спроба крос-валідації моделі на 10 частинах. Результати виявилися кращими, отже, саме ця модель була обрана фінальною. Тренувальна вибірка була перевірена на репрезентативність за допомогою показника PSI. Коефіцієнти моделі були переведені у скори відповідно до шкали FICO внаслідок чого було отримано фінальну скорингову карту, що може бути застосована у якості автоматичного фрод-скорингу.

ПЕРЕЛІК ПОСИЛАНЬ

1. Gup B. E., Kolari J. W. Commercial banking: The management of risk. Hoboken; NJ: Wiley, 2005. 323 p.

2. Neural T. Coring technologies for fighting fraud. Petersfield; Hamp- shire: Neural Technologies, 2002. 116 p.

3. Fayyad U. M., Irani K. B. Multi-interval discretization of continuous-valued attributes for classification learning. *Artificial intelligence*. 1993. № 13. P. 1022–1027.

4. Kerber R. Chimerge: Discretization of numeric attributes. *In Proceedings of the Tenth National Conference on Artificial Intelligence*. 1992. № 10. P. 123–128.

5. Molnar C. Recursive partitioning by conditional inference. *Department of Statistics:* University of Munich, 2013. P. 97-99.

6. Bhalla D. Weight of evidence and information value explained. URL: https://www.listendata.com/2015/03/weight-of-evidence-woe-and-information.html (Last accessed: 11.10.2020).

7. Zeng G. Metric Divergence Measures and Information Value in Credit Scoring. *Journal of Mathematics*. 2013. № 5. P. 1–10.

8. Burbea G., Rao C. Entropy differential metric, distance and divergence measures in probability spaces: a unified approach. *Journal of Multivariate Analysis*. 1982. № 4, P. 575–596.

9. Zeng G. Necessary condition for a good binning algorithm in credit scoring. *Applied Mathematical Sciences*. 2014. № 8. P.3229–3242.

10. Ferri C., Hernández-Orallo J., Modroiu R. An experimental comparison of performance measures for classification. *Pattern Recognition Letters*. 2009. № 30. P. 27-38.

11. Hothorn T., Hornik K., Zeileis A. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics*. 2006. № 15. P. 651–674.

12. Gashnikov M. The optimal quantization in the problem of compression of digital signals. *Computer Optics*. 2001. № 21. P. 179-185.
13. Kleinbaum D. G. Logistic regression. NY: Springer-Verlag, 1994. 282 p.
14. Hosmer D.W., Lemeshow S. Applied Logistic Regression. Wiley Publishing, Inc., 2000. 369 p.
15. Harrell F. Regression modeling strategies. NY: Springer, 2001. 608 p.
16. Satchell S. E., Xia,W. Analytic Models of the ROC Curve: Applications to Credit Rating Model Validation. *SSRN Electronic Journal*. 2007. № 3. P.76-79.

ДОДАТОК А ЛІСТИНГ ПРОГРАМИ

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from optbinning import OptimalBinning
from optbinning import BinningProcess
from optbinning import Scorecard
from optbinning.scorecard import plot_auc_roc, plot_cap, plot_ks
from optbinning.scorecard import ScorecardMonitoring
from sklearn.linear_model import LogisticRegressionCV

df_final = pd.read_csv('/Users/masha/Desktop/Diploma/fraud_final.csv')
df_last_q = df_final[df_final['date'] > '2016-10-09']
df_final= df_final.drop(columns='date')
df_last_q = df_last_q.drop(columns='date')
df_last_q = df_last_q.iloc[:,[0,1,2,3,4,5,7,8,9,10,11,12,13,14]]

x = df_final.drop('bad_flag', axis = 1)
y = df_final['bad_flag']
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 42, stratify
= y)

df_train = pd.merge(X_train, y_train, right_index = True,
            left_index = True)
df_test = pd.merge(X_test, y_test, right_index = True,
            left_index = True)

x = df_train['age'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='age', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_age = optb.transform(x, metric="woe")

x = df_train['gender'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='gender', dtype="categorical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_gender = optb.transform(x, metric="woe")
```

```
x = df_train['days_snc'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='days_snc', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_days_snc = optb.transform(x, metric="woe")

x = df_train['num_cntrp_black'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='num_cntrp_black', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_num_cntrp_black = optb.transform(x, metric="woe")

x = df_train['num_cntrp_no_hist'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='num_cntrp_no_hist', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_num_cntrp_no_hist = optb.transform(x, metric="woe")

x = df_train['balance_to_trans'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='balance_to_trans', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_balance_to_trans = optb.transform(x, metric="woe")

x = df_train['num_act_ch'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='num_act_ch', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_num_act_ch = optb.transform(x, metric="woe")

x = df_train['share_cash'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='share_cash', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
```

```
woe_share_cash = optb.transform(x, metric="woe")

x = df_train['herf_trans'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='herf_trans', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_herf_trans = optb.transform(x, metric="woe")

x = df_train['share_unknwn_purp'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='share_unknwn_purp', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_share_unknwn_purp = optb.transform(x, metric="woe")

x = df_train['share_wo_VAT'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='share_wo_VAT', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_share_wo_VAT = optb.transform(x, metric="woe")

x = df_train['share_round_trans'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='share_round_trans', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_share_round_trans = optb.transform(x, metric="woe")

x = df_train['cv_balance'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='cv_balance', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
binning_table.build()
binning_table.plot(metric="woe")
woe_cv_balance = optb.transform(x, metric="woe")

x = df_train['cv_trans'].values
y = df_train['bad_flag'].values
optb = OptimalBinning(name='cv_trans', dtype="numerical", solver="cp")
optb.fit(x, y)
binning_table = optb.binning_table
```

```
binning_table.build()
binning_table.plot(metric="woe")
woe_cv_trans = optb.transform(x, metric="woe")

woe_data = np.array([woe_age, woe_gender, woe_days_snc, woe_num_cntrp_black, woe_num_cntrp_no_hist, woe_balance_to_trans, woe_share_cash, woe_herf_trans, woe_share_unknwn_purp, woe_share_wo_VAT, woe_share_round_trans, woe_cv_balance, woe_cv_trans, y])
woe_df = pd.DataFrame(woe_data).T
woe_df.columns = ['woe_age', 'woe_gender', 'woe_days_snc', 'woe_num_cntrp_black', 'woe_num_cntrp_no_hist', 'woe_balance_to_trans', 'woe_share_cash', 'woe_herf_trans', 'woe_share_unknwn_purp', 'woe_share_wo_VAT', 'woe_share_round_trans', 'woe_cv_balance', 'woe_cv_trans', 'bad_flag']

corrmat = woe_df.corr(method = 'spearman')
plt.figure(figsize=(10,10))
sns.heatmap(corrmat, annot = True, fmt='.1g');

df_train_final = df_train.iloc[:,[0,1,2,3,4,5,7,8,9,10,11,12,13,14]]
df_test_final = df_test.iloc[:,[0,1,2,3,4,5,7,8,9,10,11,12,13,14]]

variable_names = list(df_train_final.columns[:13])
categorial = list(df_train_final.columns[1])
target = "bad_flag"

binning_process = BinningProcess(variable_names, categorical_variables= categorial)
estimator = LogisticRegression(solver="lbfgs", C=0.001)
estimator2 = LogisticRegressionCV()
scorecard = Scorecard(target=target, binning_process=binning_process,
                      estimator=estimator, scaling_method="min_max",
                      scaling_method_params={"min": 300, "max": 850})
scorecard2 = Scorecard(target=target, binning_process=binning_process,
                      estimator=estimator2, scaling_method="min_max",
                      scaling_method_params={"min": 300, "max": 850})
scorecard.fit(df_train_final)
scorecard2.fit(df_train_final)

pd.set_option('display.max_columns', None)  # or 1000
pd.set_option('display.max_rows', None)  # or 1000
pd.set_option('display.max_colwidth', -1)  # or 199
scorecard.table(style="summary")
scorecard2.table(style="summary")

y_pred = scorecard.predict_proba(df_train_final)[:, 1]
y_pred2 = scorecard2.predict_proba(df_train_final)[:, 1]
plot_auc_roc(df_train_final[target], y_pred)
plot_cap(df_train_final[target], y_pred)
plot_auc_roc(df_train_final[target], y_pred2)
plot_cap(df_train_final[target], y_pred2)
y_pred_test = scorecard.predict_proba(df_test_final)[:, 1]
plot_auc_roc(df_test_final[target], y_pred_test)
plot_cap(df_test_final[target], y_pred_test)
```

```
y_pred_test2 = scorecard2.predict_proba(df_test_final)[:, 1]
plot_auc_roc(df_test_final[target], y_pred_test2)
plot_cap(df_test_final[target], y_pred_test2)

score = scorecard.score(df_train_final)
score_test = scorecard.score(df_test_final)
score2 = scorecard2.score(df_train_final)

mask = df_train_final[target] == 0
plt.hist(score[mask], label="non-event", color="b", alpha=0.35)
plt.hist(score[~mask], label="event", color="r", alpha=0.35)
plt.xlabel("score")
plt.legend()
plt.show()

mask2 = df_train_final[target] == 0
plt.hist(score2[mask2], label="non-event", color="b", alpha=0.35)
plt.hist(score2[~mask2], label="event", color="r", alpha=0.35)
plt.xlabel("score")
plt.legend()
plt.show()

monitoring2 = ScorecardMonitoring(target=target, scorecard=scorecard2, psi_method="cart",
                                  psi_n_bins=10, verbose=True)
monitoring2.fit(df_last_q, df_train_final)
monitoring2.psi_table()
monitoring2.psi_plot()
monitoring2.psi_variable_table(style="summary")
```